



**Universidade de Brasília - UnB
Faculdade UnB Gama - FGA
Curso de Engenharia Eletrônica**

**CONVERSOR A/D BASEADO EM COMPRESSÃO POR
TRANSFORMADAS *WAVELET***

**Autor: MATHEUS FERREIRA PIMENTA
Orientador: SANDRO AUGUSTO PAVLIK HADDAD**

**Brasília, DF
2015**



MATHEUS FERREIRA PIMENTA

CONVERSOR A/D BASEADO EM COMPRESSÃO POR TRANSFORMADAS *WAVELET*

Monografia submetida ao curso de graduação
em engenharia eletrônica da Universidade de
Brasília, como requisito parcial para obtenção
do Título de Bacharelem engenharia
eletrônica

Orientador: Sandro Augusto Pavlik Haddad

**Brasília, DF
2015**

CIP – Catalogação Internacional da Publicação

Pimenta, Matheus Ferreira.

Conversor A/D baseado em compressão por transformadas *Wavelet* / Matheus Ferreira Pimenta.
Brasília: UnB, 2015. 50 p. : il. ; 29,5 cm.

Monografia (Graduação) – Universidade de Brasília
Faculdade do Gama, Brasília, 2014. Orientação: Sandro
Augusto Pavlik Haddad

1. Conversor A/D. 2. *Lipschitz*. 3. *Wavelet*. I. Haddad, Sandro
Augusto Pavlik. II. Doutor.

CDU Classificação



CONVERSOR A/D BASEADO EM COMPRESSÃO POR TRANSFORMADAS WAVELET

Matheus Ferreira Pimenta

Monografia submetida como requisito parcial para obtenção do Título de Bacharel em Engenharia Eletronica da Faculdade UnB Gama - FGA, da Universidade de Brasília, em 8 de julho de 2015 apresentada e aprovada pela banca examinadora abaixo assinada:

Prof. Dr. Sandro Augusto Pavlik Haddad, UnB/ FGA
Orientador

Dr. Heider Marconi, DFchip
Membro Convidado

Prof. José Edil Guimarães de Medeiros, UnB/ ENE
Membro Convidado

Brasília, DF
2015

Esse trabalho é dedicado aos meus pais, que entre elogios e críticas nunca deixaram de acreditar e apoiar.

“Que os vossos esforços desafiem as impossibilidades, lembrai-vos de que as grandes coisas do homem foram conquistadas do que parecia impossível ”(Charles Chaplin)

RESUMO

Conversores A/D de alto desempenho são comumente associados com uma alta dissipação de potência, devido especialmente a sua elevada taxa de amostragem necessária. Tal característica muitas vezes prejudica ou inviabiliza a utilização desejada de tais em equipamentos que requerem baixo consumo, em especial equipamentos médicos. O trabalho aqui levantado estuda e propõe um método inovador para a conversão de dados baseado na compactação do sinal por identificação de pontos importantes para a conversão e a extração do expoente de *Lipschitz* do mesmo. Ambos os pontos podem ser levantados a partir da utilização de transformadas *Wavelet*. Adicionalmente será feito um estudo sobre características intrínsecas de conversores atuais para que possa ser feita uma comparação justa com o novo método proposto. Os resultados finais mostram um desempenho comparável a conversores de desempenho elevado trabalhando em níveis de amostragem aproximadamente 100 vezes inferior, e também serão mostrados os pontos em aberto que podem ser tratados em trabalhos futuros.

Palavras-chave: *Lipschitz*. *Wavelet*. Conversores A/D. Baixa Potência. *Nyquist*

ABSTRACT

In A/D converters, high performance is commonly associated with a high power dissipation, especially due to its high sampling rate that is often required. This characteristic often impairs or prevents the desired usage of such in many devices that require a low power consumption, especially medical equipment. The work raised here studies and proposes an innovative method for data conversion based on signal compression through the identification of important conversion points and the extraction of *Lipschitz* exponents of those, both gathered by information provided from *Wavelet* transformations. A study on the intrinsic features current converters will be done so that a fair comparison with the proposed new method can be made. The final results show a great performance comparable to high resolution A-to-D converters working with sample rates about 100 times higher than the one here proposed. Also the points where the converter does not work well will be shown so it future works could use it to enhance its performance.

Keywords: *Lipschitz. Wavelet. A/D converters. Low Power. Nyquist.*

SUMÁRIO

RESUMO	VII
ABSTRACT	VIII
SUMÁRIO	IX
1 INTRODUÇÃO.....	10
1.1 Objetivos	11
1.2 Organização do trabalho	11
2 Conversores A/D	13
2.1 Princípios básicos para conversão analógico-digital	13
2.1.1 Teorema de amostragem	13
2.1.2 Filtros Anti <i>Aliasing</i>	15
2.1.3 Quantização	16
2.2 Não Linearidades e Caracterização	18
2.2.1 Caracterização estática	18
2.2.1.1 Erro de <i>Offset</i> e Erro de Ganho	19
2.2.1.2 INL e DNL	19
2.2.2 Caracterização relacionada ao espectro de frequências	22
2.3 Análise de consumo	24
3 Embasamento para o projeto do conversor	26
3.1 Transformada de <i>Fourier</i>	26
3.2 Transformada <i>Wavelet</i>	27
3.2.1 <i>Módulus Máxima</i>	29
3.2.2 <i>Vanishing Moments</i>	29
3.3 Expoente de <i>Lipschitz</i>	30
3.3.1 Obtenção do expoente de <i>Lipschitz</i> por filtros <i>Wavelet</i>	32
3.4 Reconstrução de sinal	33
4 Conversores D/A	36
4.1 Algoritmo de reconstrução	36
4.2 Bases para reconstrução	37
4.2.1 A Base polinomial	38
4.2.2 Base senoidal	39
5 Metodologia de implementação de filtros Wavelet analógicosA	41
5.1 Aproximação da Wavelet	41
5.2 Representação por espaço de estados	41
5.3 Topologia <i>Log-domain</i>	45
5.3.1 Circuitos Translineares	46
5.3.2 Operadores Logarítmicos e Exponenciais	48
5.3.3 Integrador	50
6 Circuitos Implementados	51
6.1 Aproximação dos filtros wavelet	51
6.2 Elaboração do filtro Wavelet	55
6.3 Circuito para amostragem	60
7 Resultados e performance	63
7.1 O Conversor	63
7.2 O sinal reconstituído	65
7.3 Comparação com conversores convencionais	66
7.3.1 Sinais e Figuras de mérito	66
7.3.2 Validação do desempenho para conversão de ECG	66
7.3.3 Validação do desempenho para conversão de imagens	69
7.3.4 Validação do desempenho na frequência	72
7.4 Desempenho dos circuitos	74
7.4.1 Filtros Wavelet	75
7.4.2 Circuitos de Amostragem	78
7.4.3 Cálculo do expoente de <i>Lipschitz</i>	78
7.4.4 Amostragem a partir de transformadas Wavelet	80
7.4.5 Problemáticas encontradas	80
7.5 Discussão	83
8 Conclusão	85
ReferênciasBibliográficas	87

1 INTRODUÇÃO

Conversão analógica para digital está presente em todos os lugares ao nosso redor. Em praticamente todos os equipamentos eletrônicos, a interface de conversão é responsável para conectar nosso mundo físico em forma analógica com máquinas digitais capazes de computar os dados desejados. Esses dispositivos foram os principais responsáveis pelo desenvolvimento de dispositivos eletrônicos nos últimos anos, desde a telefonia móvel até equipamentos médicos.

Quando se trata de processamento de sinais, circuitos analógicos podem se mostrar uma boa solução de baixo custo, e em alguns casos funções simples podem ser facilmente implementadas, como filtros e amplificadores. Porém, quando se trata de processos mais complexos o processamento analógico de sinais passa a se tornar mais falho, e sua funcionalidade comumente diminui e o processamento digital se mostra muito mais eficiente. Algumas das principais vantagens que o processamento digital de sinais oferece são: armazenamento de dados de forma perfeita e em quantidade elevada, adaptação de algoritmos, e elevada relação sinal ruído.(MARCEL PELGROM, 2013)

Conversores analógicos tradicionais são compostos basicamente de dois blocos, amostragem e quantização. A quantização define níveis de aproximação dentro de uma escala para convergir o valor analógico real e assim convertê-lo em níveis digitais bem definidos. Quanto maior for a divisão de níveis para a quantização, maior é a fidelidade que os valores digitais tem com relação aos seus valores analógicos. A amostragem basicamente define a quantidade de pontos que compõem o sinal que devem ser obtidos para que se possa fazer uma recriação fiel do mesmo. A amostragem dos sinais segue o princípio de *nyquist* que diz, para se recompor um sinal analógico a partir de suas amostras, a frequência de amostragem deve ser ao menos duas vezes maior que a largura de banda do sinal, mas na prática é bem mais elevada.

Devido ao aspecto da alta taxa de amostragem e da necessidade de se possuir muitos níveis de quantização para se obter uma alta fidelidade entre o sinal digital e o seu respectivo analógico, uma alta potência é associada a conversores de alta precisão. Então, desta forma, este trabalho tem como objetivo a análise de um conversor de baixo

consumo baseado em processamento analógico por filtros *Wavelet* de forma a comparar sua viabilidade com conversores comuns.

1.1 Objetivos

Devido a importância que conversores A/D tem nos tempos atuais e a necessidade que diversos sistemas tem de baixo consumo, como tags de RFID e aparelhos médicos de forma geral (como aparelhos auditivos e marcapassos), esse trabalho propõe estudar a proposta de conversor analógico-digital de baixo consumo baseado em processamento analógico proposto inicialmente em (José Alberto, 2015). Durante esse estudo propomos:

- Reproduzir o conversor a níveis de algoritmo
- Comparar figuras de mérito com outros conversores a nível de algoritmo, livres de erros e não linearidades comuns.
- Demonstrar o funcionamento do algoritmo pela conversão de imagens e sinais de ECG.
- Propor diferentes bases para a reconstrução do sinal e comparar seus resultados com os anteriores.
- Demonstrar a possibilidade de funcionamento do sistema a nível de circuito e salientar os pontos para melhorias.

Utilizaremos de simulações a nível de algoritmo sobre diferentes sinais a fim de reproduzir os resultados apresentados em (José Alberto, 2015) checando o funcionamento do conversor em questão e medir o desempenho do mesmo com relação a outros métodos de conversão comuns. A nível de circuito utilizaremos de técnicas em *log-domain* para implementar os filtros necessários e, por meio de simuladores avançados confirmaremos o funcionamento dos blocos implementados ressaltando os pontos que necessitem de melhorias até o momento.

1.2 Organização do trabalho

O trabalho aqui apresentado, está organizado de forma a construir o conhecimento no que diz respeito a conversores A/D e os métodos utilizados para o

desenvolvimento da topologia proposta visando solucionar os problemas apresentados. Para tanto o documento é dividido em 6 partes. O primeiro tópico trata da introdução do documento, para situar o leitor em relação aos aspectos atuais de conversores A/D e sua necessidade. Apresenta os objetivos gerais do trabalho e mostra como o mesmo está organizado. O segundo tópico entra a fundo nos aspectos atuais que um conversor apresenta. São apresentados os aspectos básicos de uma conversão e são tratados erros, figuras de mérito importantes para se avaliar um conversor. O terceiro tópico mostra as ferramentas básicas para o desenvolvimento do conversor proposto, em especial mostra um diagrama de blocos sobre o qual será construído um algoritmo para avaliar seu desempenho. No quarto tópico é proposto um método de reconstrução do sinal a partir dos dados obtidos, e propõe diferentes bases capazes de recompor o sinal comprimido e convertido. No quinto tópico apresentamos as bases teóricas para o desenvolvimento dos filtros wavelet utilizados, mostrando toda a metodologia para sua implementação como melhores topologias e a base para o funcionamento de circuitos translineares. No sexto tópico apresentamos os circuitos elaborados para a implementação do sistema proposto a nível de algoritmo, como os filtros *wavelet* analógicos, e amostradores. O sétimo tópico trata de comparar os resultados obtidos pelo método proposto com o método tradicional, figuras de mérito importantes são definidas e permitem dizer o quão próximo, melhor ou pior, este está com relação a métodos atuais, juntamente com a avaliação dos circuitos elaborados ao longo do trabalho e finaliza discutindo os resultados obtidos com o desenvolvimento do trabalho. No oitavo tópico concluímos mostrando vantagens e desvantagens do conversor proposto neste trabalho, e abrindo novos pontos para que trabalhos futuros dêem continuidade ao que foi feito até o momento.

2 Conversores A/D

2.1 Princípios básicos para conversão analógico-digital

Conversores analógico-digital, ou conversores A/D ou ADC, são dispositivos codificadores que convertem uma entrada analógica em uma saída digital com uma determinada precisão, ou número de bits. A figura 2.1 mostra de forma geral os blocos que compõem um conversor A/D, como filtro *anti-aliasing*, amostrador e quantizador, que serão melhor abordados nas próximas sessões.

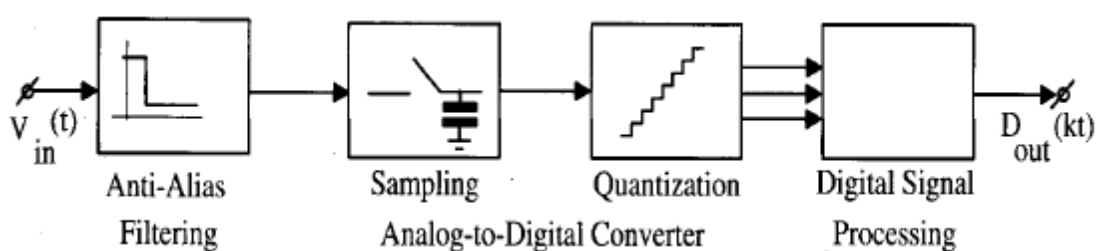


Figura 2.1 Blocos de um conversor A/D geral.(Rudy Van de Plasche,2003)

Cada bloco do conversor exerce um papel importante para a conversão, que será explicado nas sessões seguintes, como limitar a banda do sinal de entrada, amostrar os sinais de forma discreta no tempo, quantizá-lo para a geração dos valores digitais.

2.1.1 Teorema de amostragem

Em meados da década de 20, durante o estudo sobre taxa máxima de sinalização que poderia ser utilizada em um canal telegráfico de determinada largura de banda, Harry *Nyquist* acabou por estabelecer as bases para a teoria de amostragem, que anos mais tarde seria entendida por Claude E. Shannon, em 1949, para os sistemas de comunicação. O Critério de *Nyquist*, ou teorema de *Nyquist*, foi assim formulado: Se uma Função não contém frequências mais altas que BW ciclos por segundo, ela é completamente determinada dando a suas ordenadas uma série de pontos espaçados por $1/2$ BW segundos. O critério impõe uma relação simples entre a largura de banda (BW) de um sinal e a frequência de amostragem(f_s).

$$f_s > 2 * BW \quad \text{Eq. 2.1}$$

O intervalo de amostragem mínimo, $T_s=1/f_s$, fica claro com o estabelecimento do critério e deve ser escolhido caso a caso para a melhor representar um sinal. O critério é estabelecido assumindo filtros ideais e um período de tempo infinito para reconstrução do sinal que, na prática, são restrições que não são atingíveis, filtros por ventura muito complexos podem gerar distorções por fase e que para determinadas aplicações, como áudio, que podem acabar com o projeto (MARCEL PELGROM, 2013), e técnicas como *oversample* nos permitem separar banda base e banda de *alias* o suficiente para facilitar o projeto de filtros. Ao se amostrar um sinal abaixo da frequência de *nyquist*, pode-se observar um fenômeno denominado *aliasing*, onde a banda do sinal original colide com a banda base, como podemos observar na figura 2.2

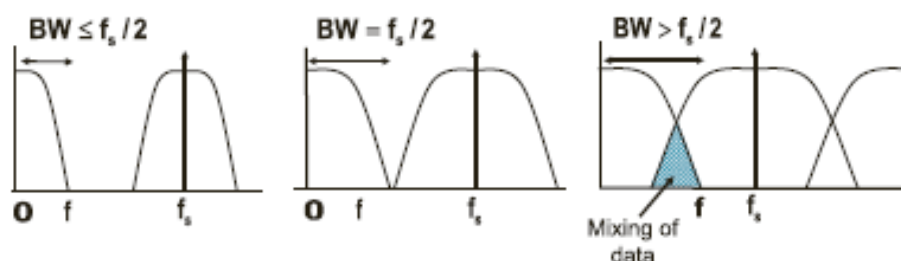


Figura 2.2 Espectro de frequência com amostragem, respectivamente, acima da frequência de *Nyquist*, igual a Frequência de *Nyquist* e menor que a frequência de *Nyquist*. (MARCEL PELGROM, 2013).

O efeito do *alias*, ou falseamento, é consequência da sobreposição da banda base com a banda de amostragem do sinal. Ocorre quando a frequência de amostragem é inferior a duas vezes a largura de banda. Esse efeito, visível no domínio da frequência, pode ser observado também no domínio do tempo, figura 2.3.

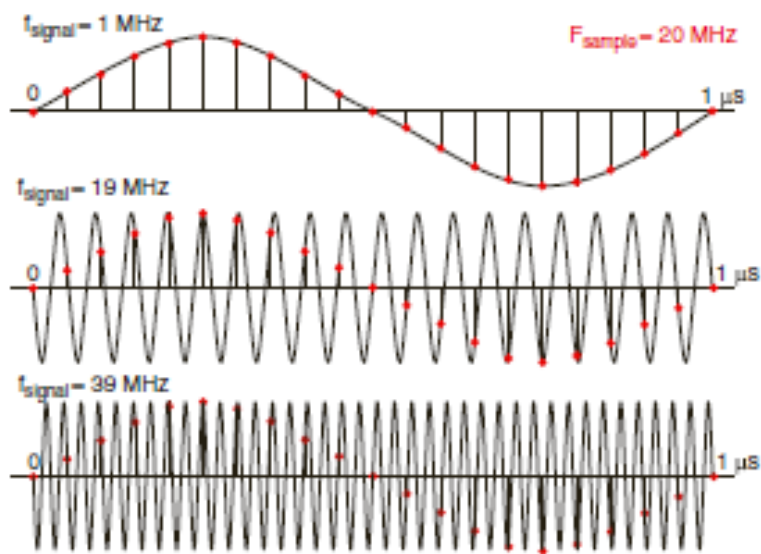


Figura 2.3 Amostragem de 3 sinais contínuos utilizando a mesma taxa de amostragem. Resultado pode gerar três dados iguais devido ao efeito de *aliasing*..(MARCEL PELGROM, 2013).

2.1.2 Filtros Anti *Aliasing*

Em diversos casos, para se limitar a frequência de amostragem, utiliza-se de filtros *anti-aliasing* para limitar a banda do sinal de entrada. De forma geral, filtros anti-aliasing são filtros passa baixa, porém em alguns casos pode-se utilizar filtros passa banda (Rudy van de Plassche, 2003). A figura 2.4 mostra com clareza a funcionalidade de um filtro *anti-aliasing*. A figura 2.4-a mostra a resposta em frequência do sinal de entrada com maior frequência de entrada igual a f_b , e é amostrada a uma frequência f_s maior que $2 \cdot f_b$ em 2.4-b. O resultado mostra um sinal que beira o ponto onde pode ocorrer o *aliasing*, e, caso a banda de f_b aumente, mesmo que pouco, já se torna impossível a recuperação do sinal. E por isso se faz necessário o uso de um filtro passa baixa para se evitar esses efeitos (Allen e Holberg, 2002).

O uso de filtros, por sua vez, pode ser algo problemático quando se deseja utilizar a amostragem na frequência de *Nyquist*. Como observado por (MARCEL PELGROM, 2013), o formato para gravação de música em CD representava uma amostragem onde a largura de banda do sinal desejado é de 20 kHz. Essa combinação deixaria uma

pequena banda de transição que varia de 20 a 24.1kHz para que o *aliasing* fosse suprimido em aproximadamente 90dB. O projeto de algum filtro para determinado projeto seria demasiado complexo e dispendioso com uma necessidade de 11 a 13 polos, além do aparecimento de não linearidades no comportamento da fase a frequências mais altas da banda base. Não linearidades que gerariam distorções na fase, e por consequência no tempo, teriam um efeito audível não aceitável para a aplicação. Em casos assim, a utilização de *oversample* ajuda muito a separar a banda base e a banda de *alias* o suficiente para aliviar muito o projeto de filtros.

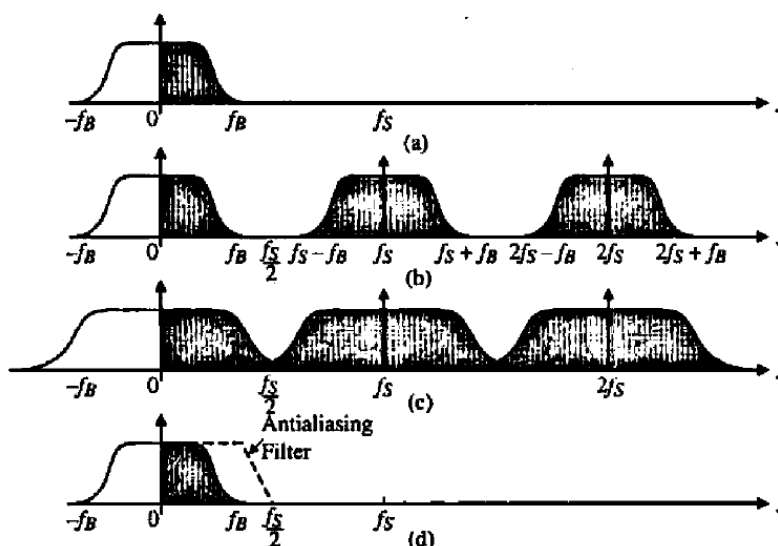


Figura 2.4 (a) Espectro de frequência do sinal contínuo. (b) Espectro de frequência com a amostragem. (c) Caso onde f_B é maior que $0.5 \cdot f_s$. (d) Uso de filtros passa baixa para evitar o *aliasing* (Allen e Holberg, 2002).

2.1.3 Quantização

Junto com o processo de amostragem, a quantização é um dos processos mais importantes na conversão de sinais. Dela surgem diversos parâmetros que caracterizam a qualidade do conversor, como não linearidades, e a partir dela que serão definidos os níveis para cada valor digital.

Quantização pode ser definida como o processo de arredondamento de um sinal analógico para sua representação digital mais próxima, dado um determinado tamanho

da palavra digital utilizada(MARCEL PELGROM, 2013). A quantização então consiste da constante comparação do sinal com valores em níveis determinados, que dividem igualmente toda a possível escala do sinal e a saída é o arredondamento do sinal ao nível mais próximo, conforme visto na figura 2.5. Dessa forma pode-se perceber que toda a conversão analógica para digital, independente da precisão utilizada, produz erros de arredondamento ligados à precisão do quantizador e esses erros por vezes são fatores limitantes para a operação do conversor.

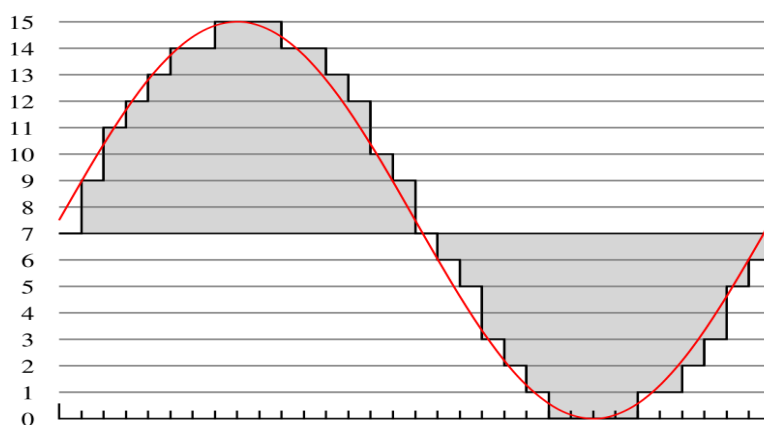


Figura 2.5 Processo de quantização de um sinal analógico

Em sistemas digitais, a base para representação analógica é o número '2' e a resolução da conversão é dada pela potência N , e assim define-se por 2^N a quantidade de níveis que definem uma escala para a quantização(Allen & Holberg, 2002). A figura 2.6 mostra uma escala de quantização com alguns de seus mais importantes dados.

Uma importante observação com relação aos níveis de quantização é a distinção entre os termos resolução e acurácia. Resolução se refere a quantidade de bits presentes na escala, enquanto a acurácia remete a qualidade a divisão de níveis dentro da quantização.

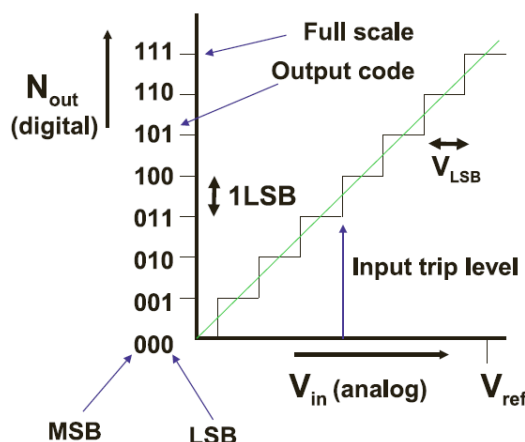


Figura 2.6 Quantizador de 3 bits padrão para conversores A/D

2.2 Não Linearidades e Caracterização

Em conversores AD, diversos fatores inserem erros irreversíveis teoria estão todos associados ao processo e quantização, porém na prática são ligados a métodos de conversão, tecnologia e topologias utilizadas, frequência de amostragem e até mesmo as habilidades dos projetistas(Rudy Van de Plassche, 2002). Os erros analisados influenciam o desempenho de conversores e levantam figuras de mérito de grande importância para a validação do dispositivo desejado. São fatores que influenciam tanto a linearidade na hora da recomposição do sinal quanto a relação entre o sinal e o ruído de quantização inserido na hora da avaliação do mesmo.

2.2.1 Caracterização estática

A caracterização estática de um conversor AD está ligada ao fato de que o sinal passa por uma quantização que não é perfeita. Esses erros podem ser observados na reta de quantização do sinal e podem ser de quatro tipos: *Gain error*(erro de ganho), *offset*(erro de compensação), *INL*(não linearidade integral) e *DNL*(não linearidade diferencial) (Allen e Holberg,2002).

2.2.1.1 Erro de Offset e Erro de Ganho

Para caracterizar esses dois erros, imaginamos uma escala de quantização de resolução infinita, uma linha. Para um caso de um conversor AD que apresenta erros de *offset*, toda essa linha é deslocada a um valor constante diferente de “zero”(assumindo que outros erros não existam).O erro de offset é então denotado como a diferença entre a escala de resolução ideal e a escala de resolução real no seu ponto de origem.

Já o erro de ganho afeta o coeficiente angular da escala de conversão, fazendo com que a escala de resolução seja mais inclinada ou menos inclinada que o ideal. Na figura 2.7 podemos ver como os dois erros são perceptíveis em escalas de quantização não ideais:

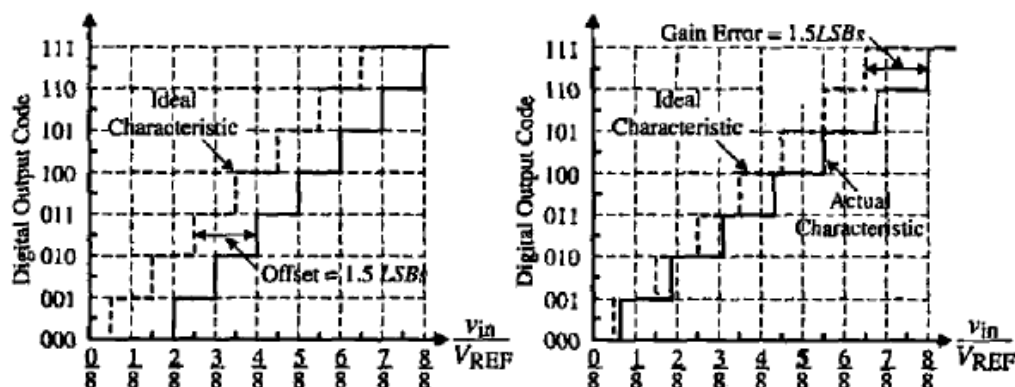


Figura 2.7 Imagem característica para medição de erros de offset(a esquerda) e erros de ganho(a direita).

2.2.1.2 INL e DNL

Não linearidades são fatores que distanciam o comportamento ideal de um conversor de uma reta. Geram distorções durante a conversão e alteram significativamente o formato de onda a ser recriado. São descritos como não linearidade integral(INL) e diferencial(DNL).

Dado um escala de resolução finita e ideal $D(n) = n * D_{LSB}$, de acordo com (MARCEL PELGROM, 2013), o INL é dado como o desvio que a escala de conversão real tem em relação a função ideal, segundo a Eq. 2.2. Em geral o INL de um conversor

é dado como uma curva, em função da posição na escala de conversão, porém, é interessante, em termos de validação, também saber o máximo valor de INL que um conversor pode obter em sua escala.

$$INL = \frac{D(n) - n * D_{LSB}}{D_{LSB}} \quad \forall n = 0 \dots (2^N - 1) \quad \text{Eq. 2.2}$$

O INL de um conversor é intimamente ligado a características dinâmicas do mesmo, como a quantidade de distorção harmônica presente na conversão. A definição dada por (MARCEL PELGROM, 2013), é aplicável em casos onde outros erros como *offset* e *ganho* não estão presentes ou são insignificantes. No caso da presença significativa de tais erros é conveniente que o INL seja calculado não a partir da ideal, mas sim da escala que melhor se ajustar ao caso real.

Outra não linearidade importante a ser caracterizada é o DNL. DNL é o desvio de cada nível de quantização com respeito ao tamanho ideal de LSB. O DNL, segundo (MARCEL PELGROM, 2013), pode ser obtido pela Eq. 2.3. Assim como o INL, o DNL também é obtido como uma função dos níveis de quantização, porém é importante também observar o máximo de DNL que o conversor desenhado pode obter.

$$DNL = \frac{D(n+1) - D(n)}{D_{LSB}} - 1 \quad \forall n = 0 \dots (2^N - 1) \quad \text{Eq. 2.3}$$

Problemas com DNL podem ser cruciais para um conversor. Podem gerar não-monotonicidade nos códigos (dois ou mais intervalos de valores analógicos apresentarem o mesmo código) e a falta de códigos (onde, ao passar de um intervalo para outro no nível analógico, ocorre a supressão de um ou mais LSBs). Na figura 2.8 podemos observar os efeitos de INL quanto DNL, incluindo os erros causados pelo segundo.

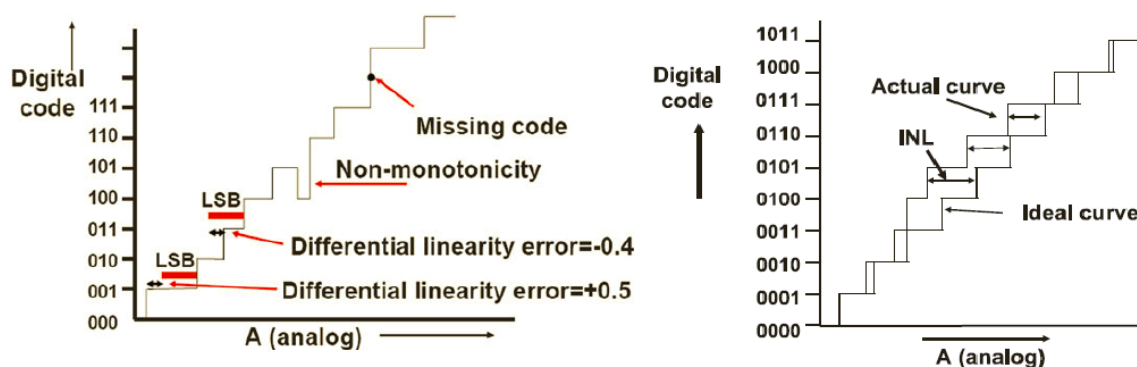


Figura 2.8 Efeitos de não linearidades nas escalas de quantização, DNL(a esquerda) e INL(a direita).

Curvas que mostram o INL e o DNL para cada escala podem ser de grande importância quando se analisa um conversor AD. Podem trazer informações que, mais do que o desempenho, podem indicar traços da arquitetura utilizada pelo conversor, em especial o INL. Na figura 2.9, por exemplo, podemos assumir, pela curva de INL do conversor, dados interessantes como uma alta distorção de segunda ordem devido ao formato de meia senóide que a distorção toma, e uma possível arquitetura que divide cada intervalo de conversão em intervalos menores (*subranging*) devido às descontinuidades em formato de dente de serra.

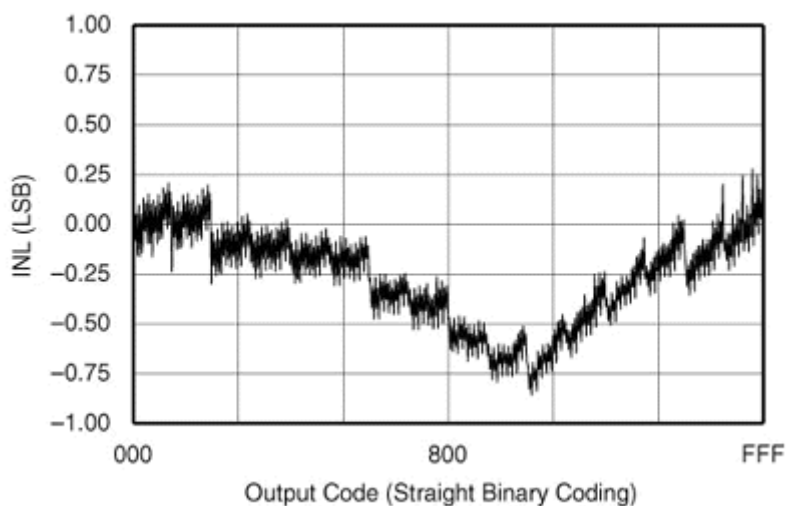


Figura 2.9 Uma curva característica de INL para um conversor de 12 bits.

2.2.2 Caracterização relacionada ao espectro de frequências

Quando tratamos de características relacionadas ao espectro de frequência, analisamos não apenas a escala de quantização como também a frequência de amostragem. Dentre os aspectos mais importantes de um conversor A/D, está sua relação sinal ruído(SNR), onde o ruído neste caso é gerado pela quantização. Essa relação nos fornece uma ideia da resolução com a qual ocorre a conversão e levanta parâmetros mínimos que determinadas resoluções devem apresentar.

A estimação da relação sinal ruído é uma medição feita a partir da função densidade de probabilidade para a conversão de um sinal analógico em uma escala de quantização perfeita. Dado um valor analógico dentro do intervalo $A_j \pm 0.5LSB$, onde A_j é o nível de quantização que o sinal está situado, o mesmo deve ser idealmente quantizado dentro do nível A_j . Ao se realizar tal aproximação, um formato de erro triangular deve ser observado, figura 2.10(Rudy Van de Plassche, 2002).

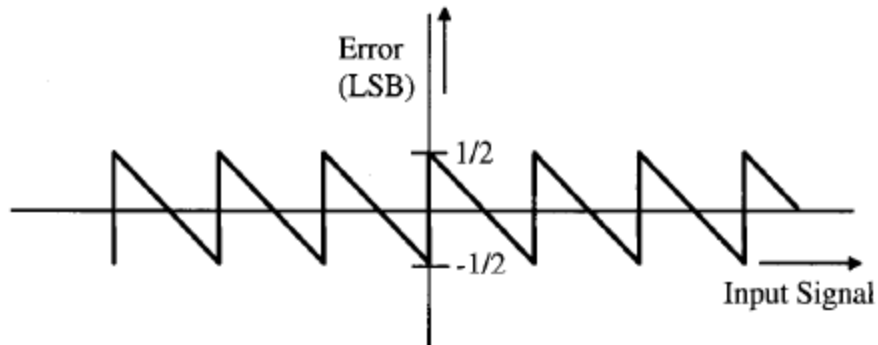


Figura 2.10 erro de quantização em uma escala perfeita

Assumindo o formato de erro, como se segue, pode-se definir a potência de quantização para um LSB como a integral da área para um LSB:

$$E(\varepsilon^2) = \int_{\varepsilon=-0.5LSB}^{\varepsilon=0.5LSB} A_{erro}^2(\varepsilon) d\varepsilon = \frac{A_{LSB}^2}{12} \quad \text{Eq. 2.4}$$

Para o cálculo da relação sinal ruído, calcula-se a potência do sinal e divide-se a mesma pela potência de quantização, onde a amplitude pode ser quantizada como $\hat{A} = 2^N A_{LSB}$.

$$Signalpower = \frac{1}{T} \int_{t=0}^{t=T} \hat{A}^2 \sin^2(\omega t) dt = \frac{\hat{A}^2}{2} = \frac{2^{2N} A_{LSB}^2}{8} \quad \text{Eq. 2.5}$$

$$SNR = \frac{Potenciadosinal}{Potenciadequantização} = \frac{3}{2} 2^{2N} \quad \text{Eq. 2.6}$$

$$SNR = N * 6.02 + 1.76 \text{ dB} \quad \text{Eq. 2.7}$$

A partir da relação sinal ruído, pode-se ter uma ideia se o conversor projetado é aplicável para a quantidade de bits que se propõe, se está acima ou abaixo do desempenho esperado.

Apesar de não parecer, de início, a relação sinal ruído de um conversor está intimamente ligada à frequência de amostragem do mesmo. A medida que se aumenta a relação da frequência de amostragem com a frequência original do sinal, a relação sinal ruído tende a se elevar também. Dado a Eq. (2.4), podemos representar a mesma como função da largura de banda que a mesma representa (Rudy Van de Plassche, 2002), como na Eq.(2.7), onde f_s é a frequência de amostragem.

$$E(\mathcal{E}^2) = \frac{A_{LSB}^2}{12f_s} \quad \text{Eq. 2.7}$$

Dada essa relação, pode-se obter a relação sinal ruído de um sinal como função da quantidade de bits e da sua frequência de amostragem como:

$$SNR = N * 6.02 - 1.25 + 10 \log_{10} \left(\frac{f_s}{2f_{signal}} \right) \quad \text{Eq. 2.8}$$

E percebe-se que taxa de amostragem elevada tende a elevar o valor da relação sinal ruído, porém, eventualmente se chega a níveis em que esse aumento passa a se tornar insignificante.

Além da relação sinal ruído, outra importante figura relacionada ao espectro de frequências é a relação sinal ruído e distorções (SINAD). Essa figura de mérito, dada pela Eq. (2.9), relaciona a potência do primeiro harmônico do sinal convertido com a potência de todos os outros harmônicos não desejados, sejam eles originados por erros de quantização ou quaisquer outros ruídos associados.

$$SINAD = 10 \log_{10} \left(\frac{Potenciado primeiro harmônico}{Potencia de todas as componentes não desejadas} \right) \quad \text{Eq. 2.9}$$

Na figura 2.11 podemos observar o ruído de quantização, as distorções harmônicas e o significado de cada figura de mérito comentada.

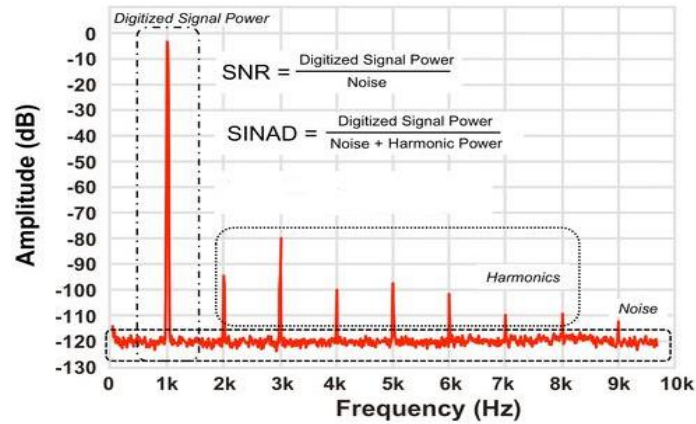


Figura 2.11 Representação do SNR e SINAD para um espectro de frequências

A partir da definição do SNR e SINAD, surge uma das mais importantes figuras de mérito para a comparação de diferentes conversores. O número efetivo de bits(ENOB), é uma figura de mérito que permite uma simples comparação da real performance de um conversor, e é dado pela Eq. (2.10), para o caso da utilização de toda a escala disponível.

$$ENOB = \frac{SINAD - 1.76}{6.02} \quad \text{Eq. 2.10}$$

O número efetivo de bits nos diz o quão próximo de um conversor ideal está o conversor projetado e devido sua simplicidade e, o fato de ser facilmente aplicável às topologias existentes, é por sua vez amplamente utilizado para compará-las.

2.3 Análise de consumo

Outro aspecto importante para a comparação de diferentes conversores, é a potência necessária para realizar uma conversão associada com a resolução do conversor desejado. Neste caso, utilizamos a figura de mérito proposta por(Gellen 2001), que define como a energia de quantização, por nível de quantização,que relaciona com a Potência disponível(P_{adc}), resolução(N) e a largura de banda do sinal(F_{BW}). Essa figura de mérito é utilizada para nos fornecer a potência mínima necessária para se realizar

uma conversão, dado o ruído térmico presente em um circuito com o mínimo de capacitores possível.

$$E_Q = \frac{P_{ADC}}{2^N 2F_{BW}} \quad \text{Eq. 2.11}$$

$$P_{min,ADC} > 48kT2^{2N} \quad \text{Eq. 2.12}$$

Essa relação é válida para conversores de *Nyquist*, como os descritos. Na prática tais valores se mostram inatingíveis nos dias atuais devido a diversas não linearidades presentes em circuitos, mas mostram possíveis limitações em projetos. Utilizando a relação da Eq.(2.11), podemos demonstrar limitações teóricas em termos de conversão, conforme a figura 2.12.

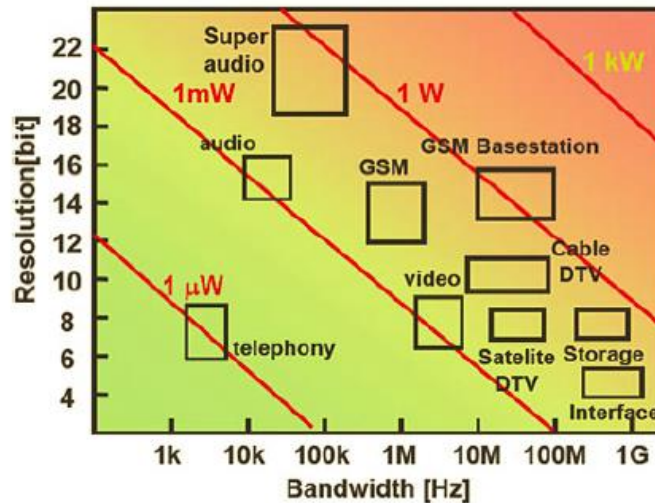


Figura 2.12 Estimação de potencia para conversores baseados em E_Q de 1pJ/(nível de quantização).

3 Embasamento para o projeto do conversor

Os seguintes tópicos tratam de funções matemáticas utilizadas como base teórica para o desenvolvimento do conversor proposto. Tal método utiliza de transformada de *Wavelet* e expoentes de *Lipschitz* para auxiliar a conversão analógico digital. Com o auxílio de tais transformadas, geradas por filtros analógicos, podemos obter resultados superiores a conversores de resolução teoricamente mais elevada e funcionando em regime de amostragem consideravelmente superior à de *Nyquist*.

3.1 Transformada de *Fourier*

A transformada de *Fourier* é uma operação matemática que nos permite analisar o espectro de frequências de um sinal desejado. A transformada de *Fourier*, por definição, é uma transformada integral da forma mostrada na Eq.(3.1), onde $S(\omega)$ é a função no domínio da frequência e $s(t)$ é a função no domínio do tempo.

$$S(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} s(t)e^{-i\omega t} dt \quad \text{Eq. 3.1}$$

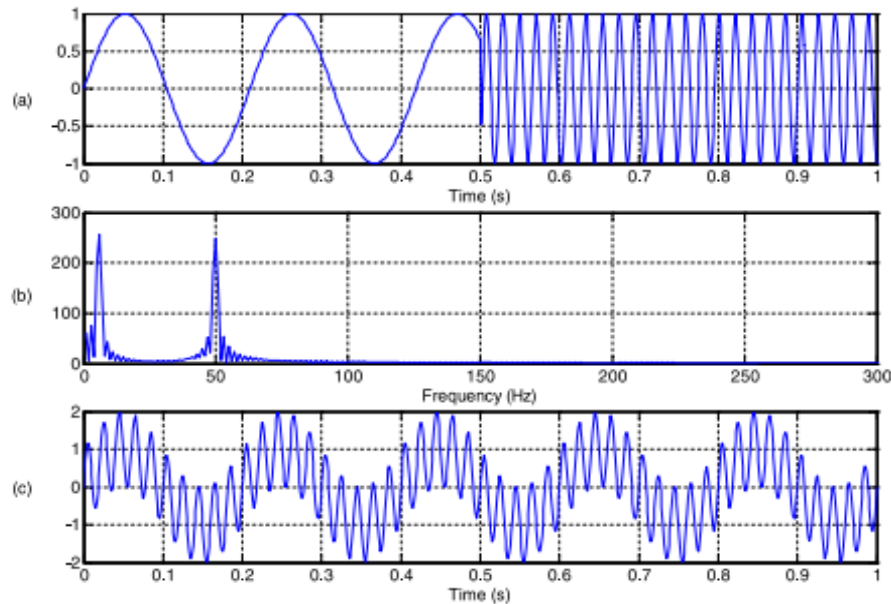


Figura 3.1 a) Sinal de entrada composto por duas frequências em instantes de tempo diferentes. b) Transformada de *Fourier* do sinal de entrada. c) transformada de *Fourier* inversa, observa-se uma recomposição errada do sinal. (Haddad and Serdjin, 2009)

A transformada de *Fourier* nos fornece os valores de amplitude de $s(t)$ no espectro da frequência, mas, para tanto, faz algumas suposições. Devido a sua base composta por senóides de duração infinita, representada por e^{-i} , é necessário que o sinal a ser analisado seja periódico e regular no tempo. Dessa forma a transformada de *Fourier* se mostra uma ferramenta ruim para a análise de sinais cujo espectro varia no tempo (Haddad and Serdjin, 2009). A figura 3.1 exemplifica a tentativa de reconstrução de um sinal não contínuo no tempo em um intervalo determinado. Como podemos observar, a transformada nos fornece um valor correto dos sinais observados no tempo, porém não nos fornece informação suficiente para a recomposição do mesmo de forma correta.

Devido a essa característica, para a análise de sinais não periódicos no tempo, é necessário a utilização de alguma outra transformada que utilize uma base com características específicas para tal.

3.2 Transformada *Wavelet*

A transformada *Wavelet* surge como uma proposta de transformada de ajuste a decomposição local em tempo e frequência. Ela se adequa bem a essas duas finalidades devido à sua característica adaptativa que nos permite, utilizando a mesma base, variar sua resolução tanto no tempo quanto no domínio da frequência.

A transformada *Wavelet* é definida pela Eq. (3.2). Podemos entender seu kernel $\psi(t)$ como um filtro passa-banda. Variando o parâmetro de escala a de $\psi(t)$ conseguimos variar a frequência central e a largura de banda do passa-banda (ALFRED MERTINS, 1999).

$$W_s(\tau, a) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} s(t) \psi^* \left(\frac{t-\tau}{a} \right) dt \quad \text{Eq. 3.2}$$

A figura 3.2 nos ajuda a entender como o funcionamento escalar do kernel da transformada *Wavelet* funciona. Para pequenos valores de a conseguimos uma ótima localização de eventos no tempo devido a característica de boa resolução para altas frequências, porém perdemos quanto a análise de frequências menos elevadas.

Entretanto, caso deseje uma análise de frequências menos elevadas, um a de maior valor proporciona uma boa análise de frequências, porém uma ruim resolução no tempo.

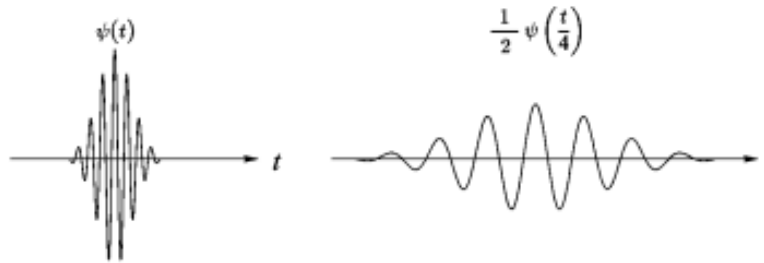


Figura 3.2 Variação do parâmetro de escala a e sua influencia no kernel da transformada

Como uma transformada, deve-se garantir que exista uma inversa capaz de recuperar toda a informação original. Para tanto, as Eq.(3.3) e (3.4) devem ser satisfeitas, onde Ψ é a transformada de *Fourier* da base ψ .

$$C_{\psi} = \int_{-\infty}^{\infty} \frac{|\Psi(\omega)|^2}{|\omega|} d\omega < \infty \quad \text{Eq. 3.3}$$

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad \text{Eq. 3.4}$$

É necessário que $|\Psi(\omega)|$ convirja rapidamente para $|\omega| \rightarrow 0$ e para $|\omega| \rightarrow \infty$. Dessa forma garantimos o comportamento de passa-banda do filtro *Wavelet*.

A relação entre resolução no tempo e resolução na frequência é fisicamente limitada, como se pode ver. Sempre ao se obter resolução no tempo perde-se resolução no domínio da frequência, assim como o contrário. Essa limitação é dada pelo princípio da incerteza de *Heisenberg*:

$$\Delta t \Delta \omega > \frac{1}{2} \quad \text{Eq. 3.5}$$

E

$$\Delta t = \sqrt{\frac{\int t^2 |\psi(t)|^2 d\omega}{\int |\psi(\omega)|^2 d\omega}} \quad \text{Eq. 3.6}$$

$$\Delta\omega = \sqrt{\frac{\int \omega^2 |\Psi(\omega)|^2 dt}{\int |\Psi(\omega)|^2 dt}} \quad \text{Eq. 3.7}$$

Onde Δt e $\Delta\omega$ são as resoluções no tempo e na frequência, respectivamente.

Algumas propriedades da transformada *Wavelet* são de extrema importância para o conversor A/D desenvolvido. O Módulo Máximo e o Vanishing moment serão discutidos nas próximas sessões.

3.2.1 *Módulus Máxima*

Propriedade de extrema utilidade para detectar singularidades em funções, chamamos módulo máxima qualquer ponto x_0 tal que $|W_s(x_0, a)| > |W_s(x, a)|$ onde x pertence a qualquer ponto na vizinhança de x_0 , seja pela esquerda ou pela direita. Dessa forma, atende-se a definição de máximos e mínimos locais, onde a Eq. (3.8) se faz válida.

Qualquer curva formada por pontos de módulo máxima no espaço de escalas é chamada linha máxima (Mallat, 1999). Essa definição nos permite encontrar pontos de máximo e mínimo locais em sinais a partir do uso de filtros *Wavelet*.

$$\frac{\partial W_s(x_0, a)}{\partial x} = 0 \quad \text{Eq. 3.8}$$

3.2.2 *Vanishing Moments*

A quantidade de vanishing moments que uma *Wavelet* possui, por definição (ALFRED MERTINS, 1999), é a quantidade de vezes que a mesma passa por zero. A definição é dada pelas Eq. (3.9) e (3.10), onde k representa a quantidade de vanishing moments e m_k representa o valor do vanishing moment k , e nos indica o quão próxima de um operador diferencial a mesma é.

$$m_k = \int_{-\infty}^{\infty} t^k \psi(t) dt \quad \text{Eq. 3.9}$$

$$m_k = (-j)^k \frac{d\Psi(\omega)}{d\omega^k} \Big|_{\omega=0} \quad \text{Eq. 3.10}$$

As propriedades de decomposição e multiresolução, que nos ajudam a comprimir sinais, são intimamente ligadas com a quantidade de vanishing moments que uma *Wavelet* possui, uma vez que fazem com que a mesma possua tal comportamento derivativo, ou passa baixas. Tem-se, pela Eq. (3.11), que para $\Psi(\omega)$ com N zeros em $\omega = 0$, a *Wavelet* $\psi(t)$ terá N vanishing moments, que significa:

$$\int_{-\infty}^{\infty} t^k \psi(t) dt = 0 \text{ para } k = 0, 1, 2, \dots, (N - 1) \quad \text{Eq. 3.11}$$

Tal propriedade nos permite encontrar pontos de máximo e mínimo em sinais a partir de sua transformada *Wavelet*, observando os pontos onde a mesma cruza o zero. Essa se mostra interessante ao fornecer bons pontos para mostrar sinais e é de grande importância na progressão do trabalho, nos fornecendo métodos para o cálculo do expoente de *Lipschitz* para a recomposição do sinal.

3.3 Expoente de *Lipschitz*

Para se caracterizar um ponto de singularidade em um sinal, é necessário quantificar muito bem sua região regular. O expoente de *Lipschitz* consegue nos fornecer informação suficiente sobre regiões regulares de sinais em determinados instantes no tempo, e em qualquer ponto de singularidade.

Define-se então f como uniformemente *Lipschitz* α sobre um intervalo $[a, b]$ se, para todo o instante $v \in [a, b]$, a seguinte relação é válida:

$$e_v = |f(t) - p_v(t)| \leq K|t - v|^\alpha \quad \text{Eq 3.12}$$

Onde e_v o erro da aproximação, K uma constante independente do ponto v , onde é feita a aproximação, e $p_v(t)$ é a aproximação de $f(t)$, em v , por meio da série de Taylor. Ou seja, se $f(t)$ é m vezes diferenciável:

$$p_v = \sum_{k=0}^{m-1} \frac{f^{(k)}(v)}{k!} (t - v)^k \quad \text{Eq. 3.13}$$

Em seu trabalho, (Haddad and Serdjin, 2009) afirmam que conhecendo o ponto onde há a ocorrência de uma singularidade é possível determinar o comportamento de uma função em um determinado intervalo de tempo. Como exemplo propõem a seguinte função para tanto:

$$s(t) = 1 - |1 - t|^\alpha \quad \text{Eq. 3.14}$$

Tal função apresenta o comportamento da figura 3.3 para diversos valores do expoente α . Atribuímos ao expoente α a característica de força da singularidade, uma vez que pode determinar o formato da função segundo a base que o rege, no caso a função $s(t)$. De acordo com (Haddad and Serdjin, 2009), observando o comportamento da função $s(t)$ em $s(1)$ conseguimos descrever com determinada precisão o valor de $s(t)$ para um determinado intervalo, então singularidades, como seu valor no tempo, suas concavidades e se valor α , são pontos que armazenam de forma compacta toda a informação para recuperar um sinal ao longo do tempo.

Tal afirmação pode ser de grande usabilidade para a conversão A/D, uma vez que atesta que, descobrindo pontos de singularidade e seus respectivos α , conseguimos informação suficiente para recuperar o mesmo. Sendo, desta forma, desnecessária a amostragem de um sinal em frequências muito superiores à de *nyquist* para que se obtenha uma conversão boa.

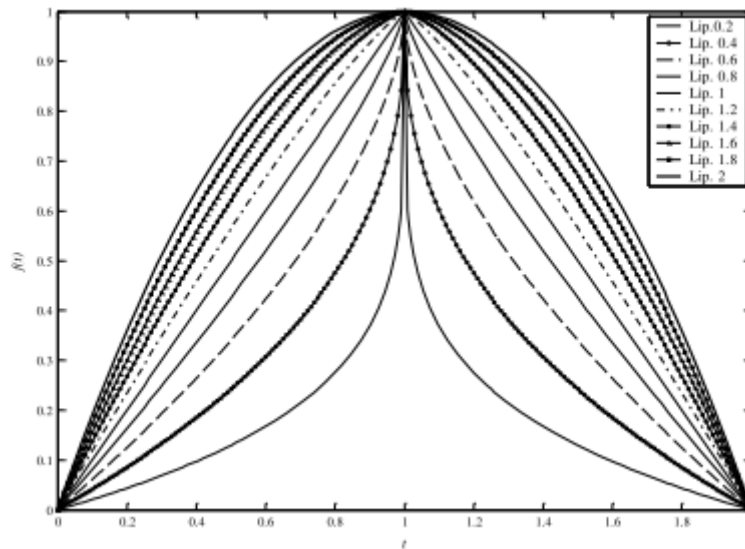


Figura 3.3 - expoentes de *Lipschitz* de 0.2 a 2 para $s(t) = 1 - |1 - t|^\alpha$ (Haddad and Serdjin, 2009)

3.3.1 Obtenção do expoente de *Lipschitz* por filtros *Wavelet*

Dado um sinal contínuo no tempo, temos que o mesmo pode ser aproximado pela Eq. (3.12). Aplicando a transformada *Wavelet* nesta equação, chegamos ao seguinte resultado graças à propriedade dos *vanishing moments*, onde temos que a transformada *Wavelet* do polinômio de aproximação é igual a zero. Esse resultado nos mostra que a transformada *Wavelet* de um sinal $s(t)$ é igual ao seu erro $e_v(t)$. E a Eq.(3.15) nos dá também uma relação entre o expoente de *Lipschitz* α e o erro de aproximação $e_v(t)$.

$$WT(s(t)) = WT(e_v(t)) + WT(p_v(t)) = WT(e_v(t)) \quad \text{Eq. 3.15}$$

De acordo com (Haddad and Serdjin, 2009), o decaimento da transformada *Wavelet* do sinal estudado, na vizinhança de v , é controlado pelo decaimento do modulus máxima incluído no cone de influência $|t - v| \leq Ca$, onde a são as escalas utilizadas e C é uma constante independente. Se garantimos que o sinal é uniformemente *Lipschitz* na vizinhança de v , então todos os *modulus máxima* dentro do cone satisfazem a Eq.(3.16).

$$|WTs(t, a)| \leq Aa^{\alpha + \frac{1}{2}} \quad \text{Eq. 3.16}$$

Tal representação pode ser linearizada por logaritmos até o formato

$$\log_2 |WTs(t, a)| \leq \log_2 A + \left(\alpha + \frac{1}{2}\right) \log_2 a \quad \text{Eq. 3.17}$$

Graças às propriedades da transformada *Wavelet* é possível então localizar temporalmente pontos de singularidade e inflexão de um sinal, e caracterizar bem uma singularidade, por meio dos logaritmos de magnitude e de escala do sinal transformado. Importante ressaltar que segundo (Mallat, 1999), uma singularidade só é definida em pontos onde o expoente de *Lipschitz* tem seu valor situado no intervalo $0 < \alpha \leq 1$, em casos fora de tal intervalo não é caracterizada uma singularidade, porém o valor do expoente ainda nos fornece informação suficiente para que o sinal seja reconstituído com fidelidade.

3.4 Reconstrução de sinal

O método aqui proposto tem como objetivo comprimir de forma que apenas informações relevantes sejam convertidas pelos métodos convencionais de conversão A/D. No modelo de compressão proposto os seguintes dados são necessários para a reconstrução a partir dos pontos máximos mínimos e singularidades do sinal:

- Posição temporal
- Amplitude
- Expoente de *Lipschitz*
- Posição temporal dos pontos de inflexão

Para a extração dos dados necessários para realizar a conversão, utilizamos de dois bancos de filtros *Wavelet*, um filtro utilizando um kernel de primeira ordem e outro um kernel de segunda ordem. O filtro de primeira ordem nos indica pontos de mínimo, máximo e singularidades, fornecendo informação suficiente para amostrar tais pontos pela sua propriedade de *vanishing moments*.

Para localização temporal de inflexões e para o cálculo do expoente de *Lipschitz* utilizamos filtros com um kernel de segunda ordem, como, por exemplo, a primeira derivada da gaussiana. Utilizando de formas de detecção dos *vanishing moments*, podemos identificar todos os pontos críticos para se reconstruir o sinal desejado, fazendo uso de apenas poucos pontos do mesmo.

Como exemplo podemos ver a análise do seguinte sinal senoidal por meio dos filtros de primeira e segunda ordem na figura 3.4. Observando essa imagem podemos observar que o banco de filtros de primeira ordem nos indica com clareza pontos de máximo e mínimo do sinal; caso houvessem singularidades, estas também seriam identificadas. Já os pontos de inflexão são bem definidos pelo banco de filtros de segunda ordem. Em ambos os casos utilizamos filtros gaussianos, “*gauss 1*” e “*gauss 2*”.

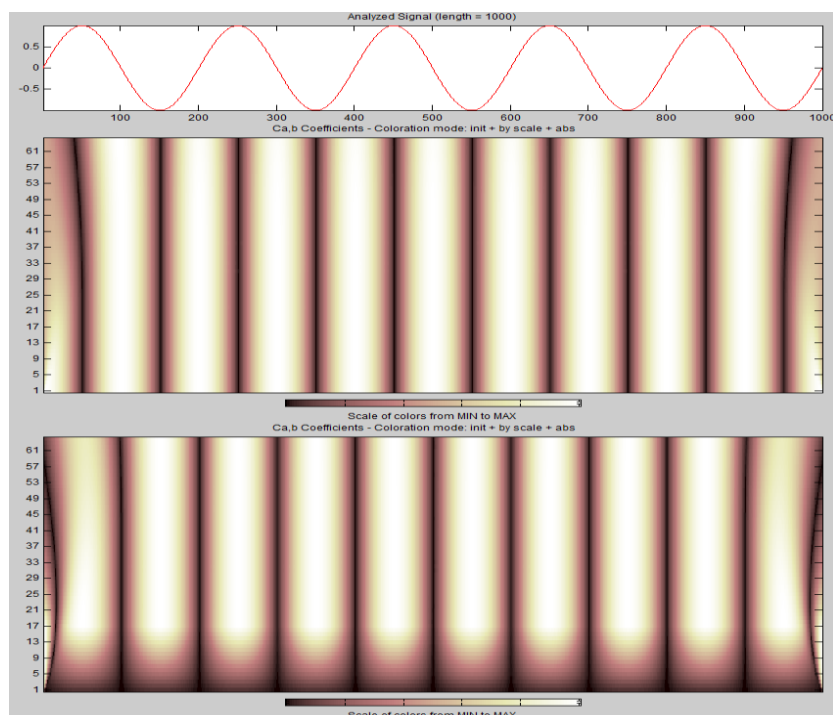


Figura 3.4 Sinal senoidal e seus coeficientes após utilização de filtros gaussianos, respectivamente , “*gaus 1*” e “*gaus 2*”.

A identificação de tais pontos é feita quando o resultado de saída do filtro cruza o valor de zero. Os pontos de máximo e mínimo nos dão uma referência de amplitude para a aproximação de *Lipschitz*, enquanto os pontos de inflexão nos fornecem informação suficiente para o intervalo de aproximação onde o valor de *Lipschitz* extraído é válido. A definição de mínimos e máximos é feita iterativamente pelo algoritmo de reconstrução, que determina máximos e mínimos locais comparando o valor da amostra em questão com seus vizinhos diretos.

A figura 3.5 mostra uma possível implementação em nível de diagrama de blocos que o possível conversor pode ter:

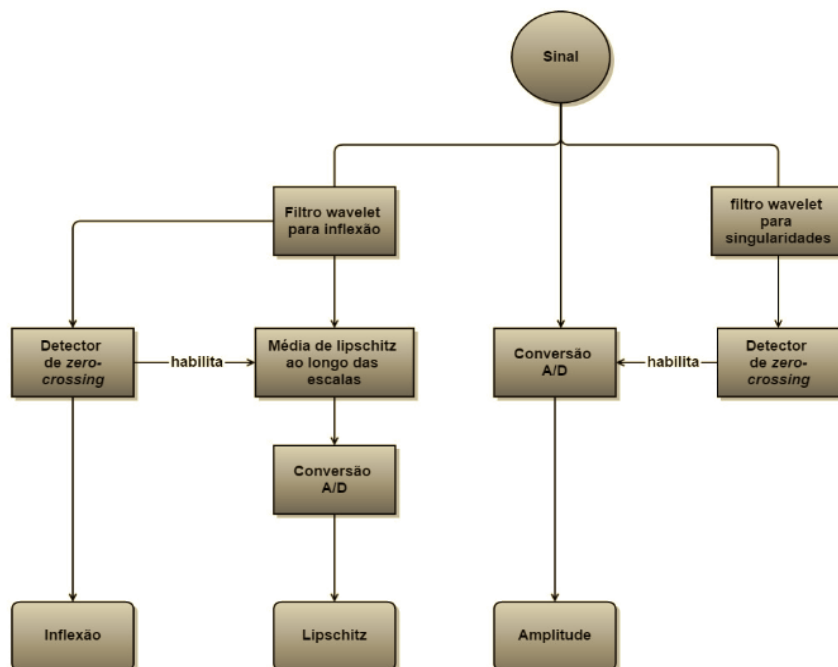


Figura 3.5 Modelo em nível de diagrama de blocos

4 Conversor D/A

Tão importante quanto a amostragem do sinal é saber como reconstruir o sinal estudado. Uma vez que o método sugerido é uma compactação, se faz necessário um método para inverter essa transformação compacta e reconstruir o sinal estudado. No tópico 4.1.1 mostramos como funciona o algoritmo utilizado para reconstruir a informação.

4.1 Algoritmo de reconstrução

O algoritmo proposto nessa sessão baseia-se em uma aproximação polinomial demonstrada em (Anton & Rorres 2001), também chamada de aproximação pela spline cúbica. Tal método nos permite interpolar dois pontos a partir de uma base, ou função, que nos forneça informação o suficiente sobre como deve ser o comportamento da função entre o intervalo determinado por tais pontos.

Para realizarmos a interpolação definimos algumas condições iniciais, um intervalo $[a,b]$ a ser interpolado e um terceiro ponto, aqui chamado de ponto de conformação (*CM*) que garante a continuidade entre as funções usadas para interpolar os pontos do intervalo $[a,b]$. As funções utilizadas devem seguir as seguintes condições para que a função que interpola os dois pontos seja contínua, e satisfaça as condições iniciais:

- $f_1(a) = A_1$ e $f_2(b) = A_2$ onde A_1 e A_2 são os valores de amplitude para os pontos a e b respectivamente, e f_1 e f_2 são, respectivamente, as funções utilizadas para a interpolação entre $[a,CM]$ e $[b,CM]$.
- $\frac{df_1}{dt} = \frac{df_2}{dt}$ de forma a garantir a continuidade das funções ao longo do intervalo $[a,b]$.
- f_1 e f_2 devem ser capazes de relacionar o formato da função pelo seu expoente de *Lipschitz*

O ponto de conformação, no caso do conversor, é o ponto de inflexão entre as duas amplitudes do extremo do intervalo que desejamos interpolar. Como os dois pontos

a se interpolar representam, em todos os casos, um ponto de máximo e mínimo local, se faz necessário inverter a concavidade da função base utilizada, e o ponto de inflexão nos diz o momento em que essa inversão deve ocorrer, e, dessa forma, se mostra essencial para a reconstrução precisa do sinal. Na figura 4.1, pode-se observar, em linhas gerais, como o método proposto.

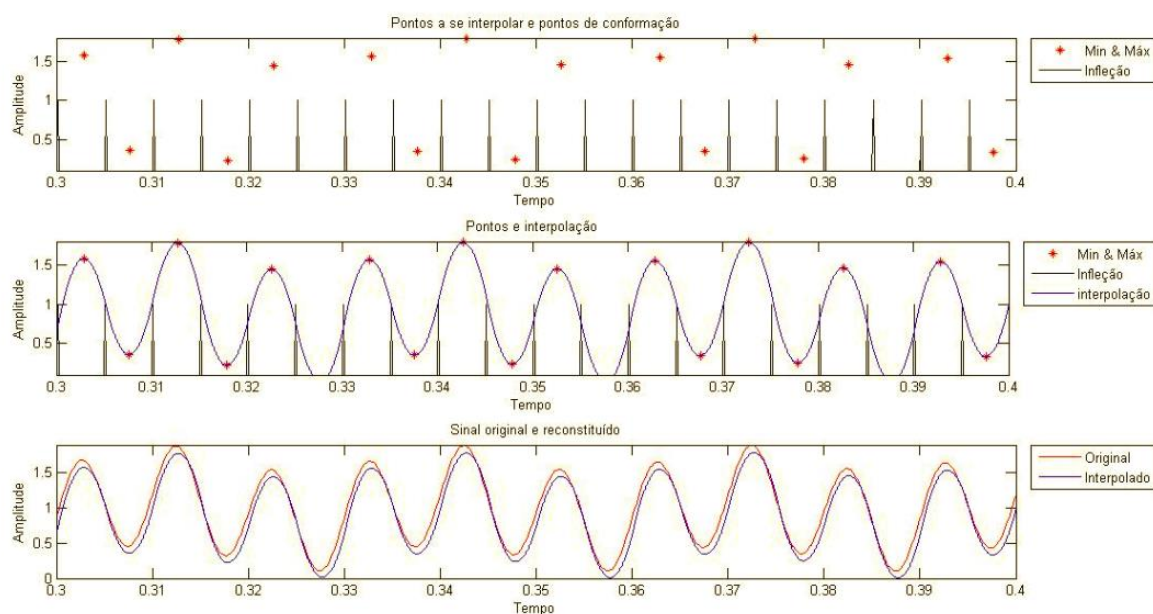


Figura 4.1 Pontos a se interpolar forma de interpolação. Base polinomial utilizada, ver 4.2.1

Importante notar na figura 4.1 que apenas o instante de tempo onde ocorrem as inflexões é necessário para aplicar o método proposto. Logo, não é preciso utilizar nenhum método de conversão para os pontos de inflexão, apenas sinalizar o momento em que os mesmos ocorrem.

4.2 Bases para reconstrução

Foram feitos estudos para utilização de diferentes bases que poderiam ser utilizadas no método de interpolação proposto e servir melhor aos resultados do conversor. Tais bases caracterizam como o algoritmo aproxima dois pontos convertidos e a qualidade como isso é feito está atrelado à melhoria de desempenho do conversor.

Duas bases foram então levantadas e serão discutidas nos tópicos a frente, a base polinomial e a base senoidal

4.2.1 A Base polinomial

Baseado no polinômio da Eq. (3.14) de (Haddad e Serdijn, 2009), a função proposta na Eq. (4.1) tem como papel aproximar um ponto amostrado em um intervalo, tendo conhecimento do valor do expoente de *Lipschitz*. A adaptações feitas à Eq. (3.14) garantem ao polinômio uma característica adaptativa, sendo possível ajustar o intervalo que o mesmo irá interpolar, amplitude variável, e deslocamento temporal.

$$f(t) = \frac{A}{v^\alpha} (v^\alpha - |\tau - t|^\alpha) \quad \text{Eq. 4.1}$$

No caso, τ representa o deslocamento temporal, A representa a amplitude do sinal, v é a dilatação temporal que o sinal representa e α o expoente de *Lipschitz* extraído no ponto de interpolação. Com esses ajustes é possível localizar a função em qualquer ponto no espaço cartesiano, e a torna ajustável para a nossa representação. É possível compreender com mais clareza a influência de cada parâmetro observando a representação da função para $\tau = 1$, $v = 0.4$ e $A = 2$ com $\alpha = [0.8; 3]$ na figura 4.2

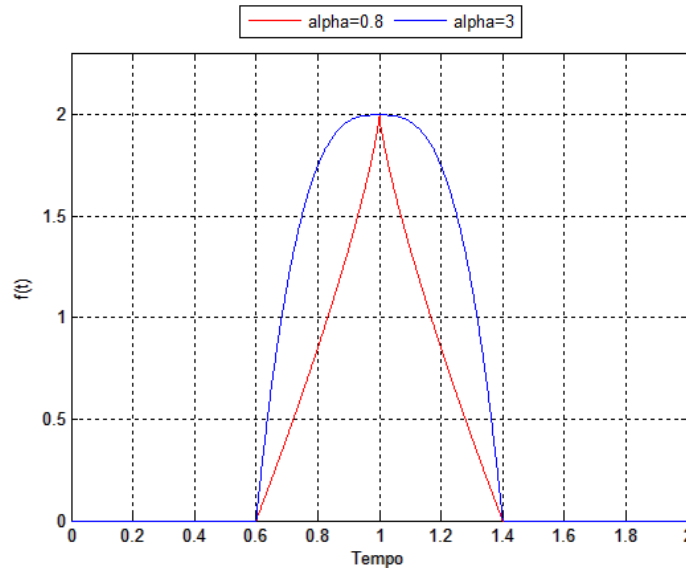


Figura 4.2 Equação(4.1) em um determinado set de parâmetros, $\tau = 1$, $v = 0.4$, $A = 2$ com $\alpha = [0.8; 3]$

O parâmetro v é fornecido então pelo ponto de inflexão, tal que v é a distância entre o ponto a e o ponto de conformação do algoritmo, neste caso dado pela localização temporal da inflexão. Logo, para o ponto a $v = |a - CM|$.

4.2.2 Base senoidal

A ideia de utilizar uma base senoidal surge do fato da maior parte dos sinais convertidos serem de forma periódica e não apresentarem descontinuidades, sendo representadas por elementos da série de *Fourier*. A partir da ideia, chegou-se a seguinte função:

$$f(t) = \frac{A}{D} (D - |\sin(2\pi\phi_{(\alpha,v)}t + J_{(\tau,v)}\tau)|^\alpha) \quad \text{Eq. 4.2}$$

Onde A representa a amplitude do sinal amostrado, τ representa o deslocamento no eixo das abscissas, α é o expoente de *Lipschitz* e $\phi_{(\alpha,v)}$ e $J_{(\tau,v)}$ são funções que têm como objetivo linearizar o deslocamento τ e a dilatação da função, componente v da base polinomial(ver 4.2.1). D no caso representa uma modulação da amplitude.

$$\phi_{(\alpha,v)} = \frac{\arcsin(D^{1/\alpha})}{2\pi v} \quad \text{Eq. 4.3}$$

$$J_{(\tau,v)} = \left(\frac{\pi - 2\pi\phi_{(\alpha,v)}\tau}{\tau} \right) \quad \text{Eq.4.4}$$

Como ilustração do funcionamento da base polinomial, temos a figura 4.3. Análoga à figura 4.2, temos $D = 1, \tau = 1, v = 0.4, A = 2$ com $\alpha = [0.8; 3]$ sendo representado.

Pode-se perceber que existe alguma diferença entre as duas formas de aproximação, porém, ao se comparar a aproximação da base senoidal a respeito de sinais formados por formatos de onda senoidal com a aproximação pela base polinomial, fica claro que a base senoidal é mais apropriada para esses casos.

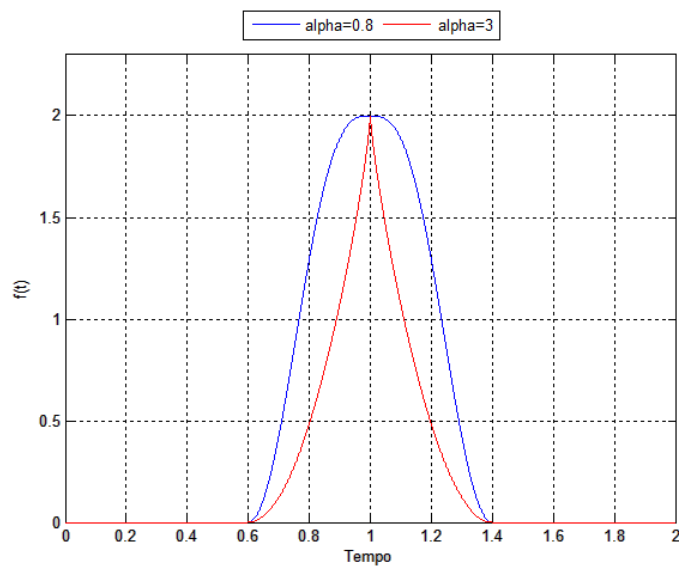


Figura 4.3 Equação(4.2) em um determinado set de parâmetros, $D = 1$, $\tau = 1$, $\nu = 0.4$,
 $A = 2$ com $\alpha = [0.8; 3]$

5 Metodologia de implementação de filtros Wavelet analógicos

Essa seção tem como objetivo descrever a metodologia apresentada por *Haddad e Serdjín* para a implementação de filtros Wavelet analógicos. Tais filtros implementados são de grande importância para a amostragem do conversor e também para a aproximação do coeficiente de lipschitz. O método apresentado gera um filtro wavelet para cada base e escala desejadas, caso haja a necessidade de uma implementação de novas escalas da mesma base devemos gerar um novo filtro(Haddad and Serdjín, 2009).

5.1 Aproximação da Wavelet

Um método simples para aproximação de funções é a técnica desenvolvida por *Henry Padé*. O método aproximação de Padé funciona como uma aproximação de uma função qualquer $F(x)$ por meio de um polinômio de ordem m/n onde m e n são inteiros com $m \geq 0$ e $n \geq 1$ conforme descrito na equação 5.1. Tal aproximação é feita a partir dos coeficientes de Taylor da função original.

$$f(x) = \frac{\sum_{j=0}^m a_j x^j}{1 + \sum_{i=0}^n a_i x^i} \quad \text{Eq. 5.1}$$

Tal método é interessante para se implementar a transformada de Laplace de qualquer função, uma vez que seu resultado nos retorna diretamente uma divisão de polinômios pronta para a implementação(Haddad and Serdjín, 2009). Tal função de transferência possuirá deve possuir um número de polos superior ao número de zeros de forma a manter a causalidade do filtro a ser implementado. O método adotado então consiste em representar a resposta desejada do filtro a ser utilizado no domínio da frequência e seguida encontrar sua aproximação pelo método de *Padé*. Com tal representação é possível se utilizar da representação por espaço de estados para a implementação do filtro desejado.

5.2 Representação por espaço de estados

A partir da função de transferência obtida pela aproximação de *Padé* podemos descrever o filtro em uma representação de espaço de estados. A descrição de espaço

de estados não é única e permitem que o filtro possa ser implementado de forma a suprir de melhor forma os requisitos desejados do filtro. Com a variação dos coeficientes podemos implementar diferentes topologias que tem diferentes performances.

A representação por espaço de estados é uma forma de se representar equações diferenciais de segunda ordem ou superior em um sistema linear de equações diferenciais de primeira ordem. Vemos na equação 5.2 a forma mais comum de se representar um sistema em espaço de estados. As entradas e saídas são representadas pelos vetores $u(t)$ e $y(t)$ respectivamente. A matrix $x(t)$ representa os estados do sistema e deve ter tamanho $n \times n$ onde n é a ordem da equação diferencial que define o sistema.

$$\begin{aligned}\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t)\end{aligned}\tag{Eq. 5.2}$$

Cada uma das Matrizes, A B C e D , definem os coeficientes que implementam a função de transferência desejada por meio de seus coeficientes. A Figura 5.1 mostra a representação por espaço de estados em formato de diagrama de blocos.

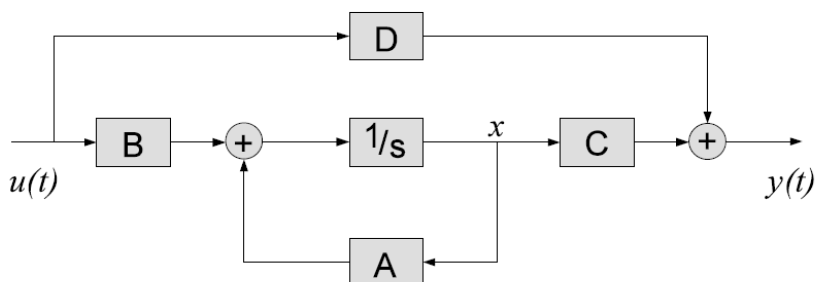


Figura 5.1 Representação de espaço de estados em diagrama de blocos

O filtro a ser implementado, devido à aplicação para baixo consumo, deve ser o menos sensível ao ruído o possível. Nesse caso a análise da sensibilidade ao ruído vem diretamente do DNR (Dynamic Range). O DNR é definido como a relação entre a maior amplitude de sinal a ser processado e o ruído interno gerado pelo circuito. O DNR de um filtro pode ser otimizado pelas matrizes de Observabilidade e Controlabilidade de um sistema por meio da equação 5.3. Deve-se encontrar o menor fator Fdr possível, onde todos os elementos podem ser escolhidos pelo projetista().

$$Fdr = \frac{\max K_{ii}}{(2\pi)^2} \sum_i \frac{\alpha_i}{C_i} \omega_{ii} \quad \text{Eq. 5.3}$$

Onde K_{ii} e ω_{ii} são os elementos diagonais das matrizes de Controlabilidade e Observabilidade respectivamente, α_i é a soma absoluta dos elementos da i -ésima linha de A e C_i são os i -ésimos elementos do vetor de saída C().

Outro elemento importante para a escolha de uma topologia de filtro a ser implementada é a quantidade de elementos que o irão compor. Para um circuito de baixo consumo é ideal que utilizemos a menor quantidade possível de componentes. Podemos estimar a quantidade de componentes a serem utilizados pelas entradas das matrizes A, B e C diferentes de zero, tal propriedade se chama *sparsity*. Matrizes com alto número de elementos diferentes de zero possuem uma elevada *sparsity*. Cada elemento das matrizes do espaço de estados será mapeado em conexões em forma de circuitos de acordo com a topologia utilizada. Cabe ao projetista encontrar a topologia que proporciona o melhor *Trade-off* entre os aspectos desejados e a quantidade de elementos.

No caso de baixo consumo, de acordo com o estudo produzido por (Haddad and Serdjin, 2009) a topologia que apresenta melhor *Trade-off* entre *sparsity* e *DNR* é a *Orthonormal*. Alguns outros pontos positivos da utilização de tal topologia são citados como a habilidade de se implementar qualquer função de transferência estável e devido à forma que é concebida, a topologia é mais fácil de se implementar do que outras que apresentam bons resultados de *sparsity* e *DNR*. Vemos abaixo a estrutura das matrizes A, B e C do espaço de estados do sistema.

$$A = \begin{bmatrix} 0 & \alpha_1 & \dots & & 0 \\ -\alpha_1 & 0 & \alpha_2 & \dots & 0 \\ & -\alpha_2 & 0 & \alpha_3 & \dots \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & & \dots & -\alpha_{n-2} & 0 & \alpha_{n-1} \\ 0 & & \dots & & -\alpha_{n-1} & \alpha_n \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \sqrt{\frac{\alpha_n}{\pi}} \end{bmatrix}, \quad C^T = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ \vdots \\ c_{n-1} \\ c_n \end{bmatrix}$$

Figura 5.2 Representação em espaço de estados da topologia orthonormal

No estudo apresentado em (Haddad and Serdjin, 2009), comparou-se diversos topologias diferentes para implementação de filtros pela representação de espaço de estados. Vemos na Figura 5.3 a diferença de desempenho de cada topologia no que tange o *DNR*, por meio do fator *Fdr*. Na comparação apresentada vemos que a topologia *orthonormal* apresenta o melhor desempenho e o mais próximo do ideal apresentado desempenho próximo à representação ótima de espaço de estados. No mesmo estudo chegou-se à conclusão de que à medida que a ordem do filtro aumenta, a diferença de desempenho também aumenta. Enquanto para filtros de segunda ou terceira ordem vemos pouca diferença no desempenho, para filtros de ordens elevadas, sexta em diante, a diferença se torna significativa.

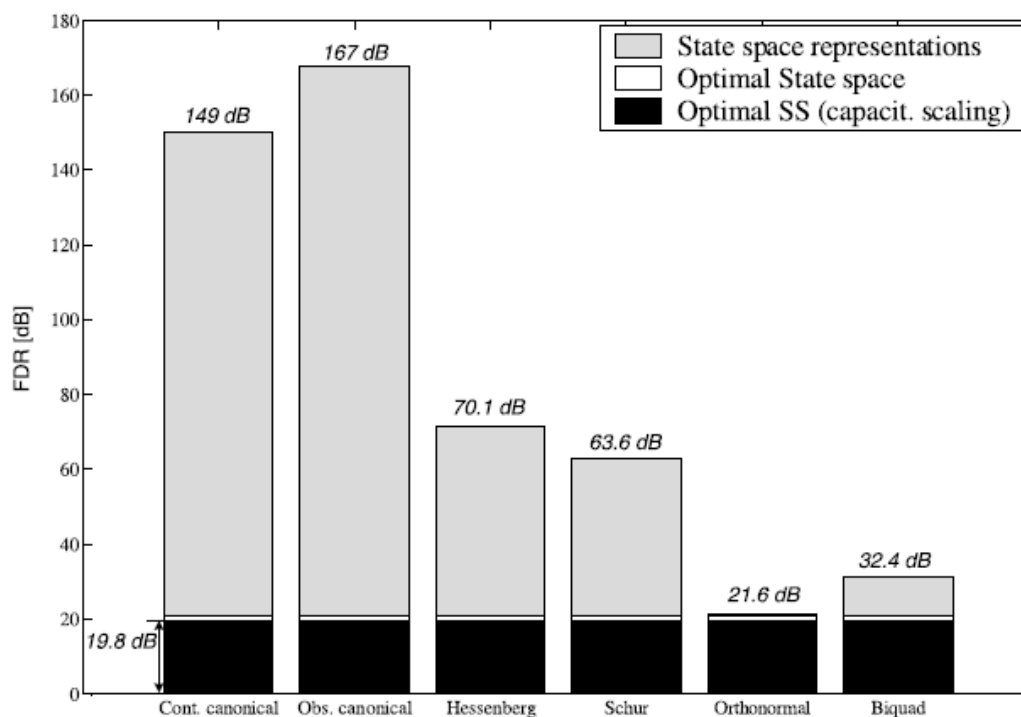


Figura 5.3 Comparação das Fdrs de diferentes descrições em espaço de estados da função *Morlet* por aproximação de padé[8/10].

5.3 Topologia *Log-domain*

A topologia de filtros operando em *Log-domain* foi primeiramente introduzida em 1979 pela *Audio Engineering Society* e desenvolvido por Adams (Gordon & Vincent, 2002) ao perceber a possibilidade de se utilizar a combinação de diodos e capacitores ao invés de filtros formados por resistores e capacitores. Pelo controle da corrente de *bias* do diodo era possível se controlar a frequência de corte do filtro desejado. O mesmo definiu os filtros em *log-domain* como: “Um circuito composto por elementos lineares e não lineares, os quais, quando combinados entre um conversor logarítmico e anti-logarítmico farão com que o sistema se comporte como um filtro linear” (R. W. Adams, 1979). Vemos na figura 5.4 a representação em diagrama de blocos da afirmação acima.

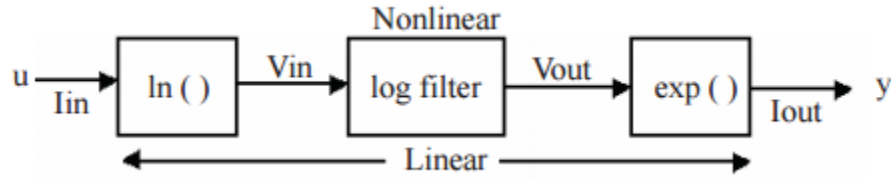


Figura 5.4 Representação do princípio de operação de filtros *log-domain*

5.3.1 Circuitos Translineares

Circuitos translineares são aqueles baseados na relação exponencial entre tensão e corrente de qualquer dispositivo, e possuem tanto como sinais de entrada quanto de saída operando no modo corrente. Dispositivos que apresentam tais características são diodos, transistores bipolares e MOSFETs operando na região de inversão fraca. Devido a tais características, circuitos translineares são capazes de realizar uma grande quantidade de operações algébricas (Gordom & Vincent, 2002). Tais circuitos tendem a trabalhar de forma independente à magnitude dos sinais de entrada, tendo uma saída dependente das razões entre elas. Podemos expressar a relação entre tensão e corrente para transistores bipolares na forma da equação 5.4.

$$V_{BE} = V_T \ln \left(\frac{I_c}{I_s} \right) \quad \text{Eq. 5.4}$$

Em um Loop simples representado pela figura 5.5 podemos observar uma das principais características dos circuitos translineares. Loops translineares são loops de tensão de base-emissor para transistores bipolares. Em tais loops a seguinte relação se mostra verdadeira.

$$V_{BE1} + V_{BE3} = V_{BE2} + V_{BE4} \quad \text{Eq. 5.5}$$

$$V_T \ln \left(\frac{I_{c1}}{I_s} \right) + V_T \ln \left(\frac{I_{c3}}{I_s} \right) = V_T \ln \left(\frac{I_{c2}}{I_s} \right) + V_T \ln \left(\frac{I_{c4}}{I_s} \right) \quad \text{Eq. 5.6}$$

Aplicando o operador exponencial e considerando V_T de todos os transistores muito próximo chegamos a relação da equação 5.7:

$$i_1 i_3 = i_2 i_4 \quad \text{Eq. 5.7}$$

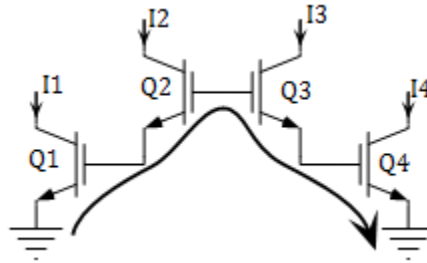


Figura 5.5 Loop translinear

Tais Loops, também chamados de Loops translineares estáticos (STL), são uma das maiores características dos loops translineares e permitem a implementação de diversas funções de transferência independentes da frequência. Tais loops são sempre implementados por um número par de transistores onde a quantidade de dispositivos orientados no sentido horário deva ser sempre igual à quantidade de dispositivos orientados no sentido anti-horário (Haddad and Serdjin, 2009).

Por sua vez, pode-se implementar funções de transferências dependentes da frequência, ou não lineares, por meio de loops translineares dinâmicos (DTL), exemplificado na figura 5.6. Aplicando a derivada à equação, estabelecemos a relação:

$$I_c = I_s e^{\left(\frac{V_{cap}}{V_T}\right)} \quad \text{Eq. 5.8}$$

$$i_c = \frac{d}{dt} \left(I_s e^{\left(\frac{V_{cap}}{V_T}\right)} \right) \quad \text{Eq. 5.9}$$

$$i_c = \frac{1}{V_T} I_c \frac{d}{dt} (V_{cap}) \quad \text{Eq. 5.10}$$

Pela equação da tensão no capacitor podemos definir a corrente do capacitor pela derivada da corrente de saída $I_c = I_{out}$.

$$C V_T i_{out} = I_{out} I_{cap} \quad \text{Eq. 5.11}$$

A combinação do Loop estático com a característica dinâmica inserida pelo capacitor podemos definir os circuitos DTL como o mapeamento da variação temporal

de uma corrente em um produto de outras correntes. O uso de circuitos translineares dinâmicos abre a possibilidade uma série de classes de filtros, como os filtros operando em *log-domain*.

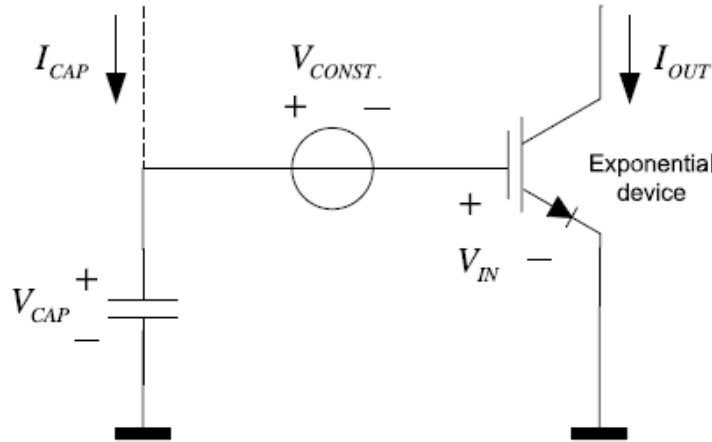


Figura 5.6 Princípio do DTL (Haddad and Serdjin, 2009).

5.3.2 Operadores Logarítmicos e Exponenciais

É possível utilizar as propriedades logarítmicas e exponenciais de circuitos de forma interessante para circuitos *low-voltage*. Se utilizar da propriedade logarítmica para a compressão dos sinais e a exponencial para expansão do mesmo é o princípio básico para a operação de filtros em *log-domain* e uma vez que distorções e ruídos que prejudicam o *dynamic range* do circuito são adicionadas em um domínio comprimido do sinal, os mesmos são menos prejudiciais após a expansão do mesmo. Tal princípio de compressão e expansão é chamado *companding*. Definiremos neste momento sinais comprimidos e expandidos como sinais em *log domain* e lineares respectivamente. De acordo com (Gordom & Vincent, 2002) podemos definir tais operadores matemáticos como:

$$LOG(x) = 2V_T \ln \left(\frac{I_0 + x}{I_0} \right) \quad \text{Eq. 5.12}$$

$$EXP(x) = I_0 e^{\frac{x}{2V_T}} - I_0 \quad \text{Eq. 5.13}$$

Exemplificando tais operadores em circuitos vemos a figura 5.7

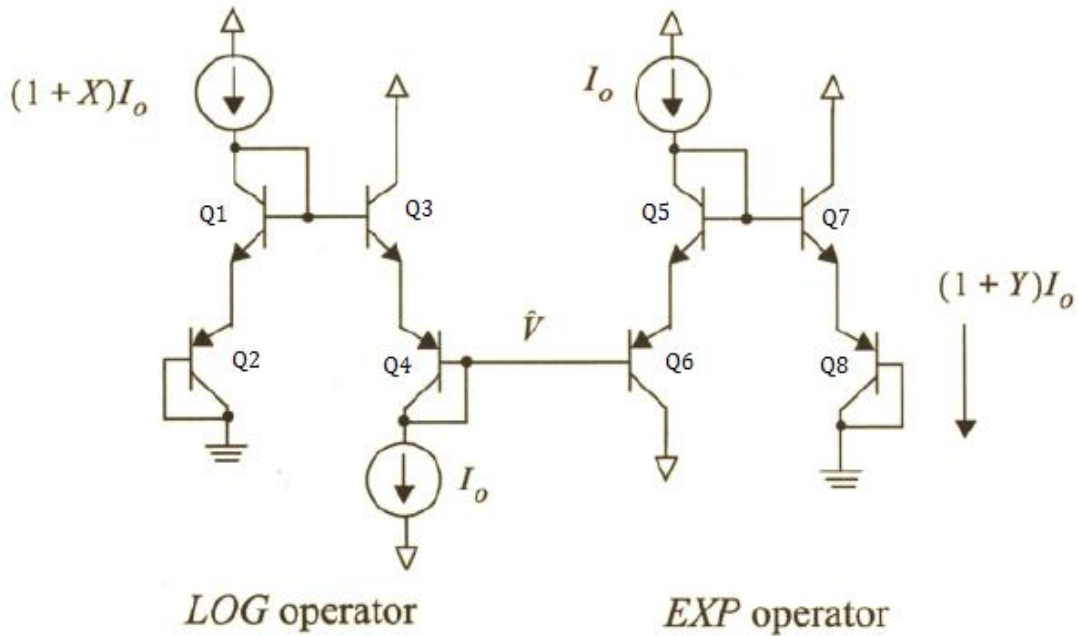


Figura 5.7 Operadores logarítmicos e exponenciais

Aplicando o loop estático no operador logarítmico vemos que :

$$I_2 I_4 = e^{\frac{\hat{V}}{V_t}} I_1 I_3 \quad \text{Eq. 5.14}$$

$$(1 + x) = e^{\frac{\hat{V}}{2V_t}} \quad \text{Eq. 5.15}$$

$$\hat{V} = 2V_t \ln(1 + x) \quad \text{Eq. 5.16}$$

Realizando a volta pelo circuito exponencial vemos que

$$I_6 I_8 = e^{\frac{\hat{V}}{V_t}} I_5 I_7 \quad \text{Eq. 5.17}$$

$$(1 + Y) = e^{\frac{2V_t \ln(1+x)}{2V_t}} \quad \text{Eq. 5.18}$$

$$(1 + Y) = (1 + X) \quad \text{Eq. 5.19}$$

No circuito apresentado \hat{V} é a tensão comprimida em *Log domain* enquanto os sinais são representados por XI_o e YI_o e para o caso de $X=Y$ temos que $EXP(LOG(x)) = x$ graças à natureza inversa entre operadores logarítmicos e exponenciais.

5.3.3 Integrador

O integrador utilizado segue a topologia apresentada na figura 5.8 Tal integrador possui terminais para introdução de correntes positivas e negativas e portanto é classificado como classe AB. O par Q1 e Q2 trabalham injetando carga no capacitor formando a entrada positiva enquanto o par Q3 e Q4 trabalha removendo carga do mesmo por meio do espelho Q7 e Q8 atuando como entrada negativa. Tal estrutura é muito comum em todos os integradores *log domain* apresentados em literaturas. Q5 e Q6 formam buffer de corrente para a tensão V_o . Dessa forma a tensão de saída V_o aparece igualmente nos emissores de Q2 e Q4. E podemos escrever as correntes do *log domain* como:

$$I_{c2} = (I_0 + I_{ip})e^{\frac{V_{ip}-V_o}{V_t}} \quad \text{Eq. 5.20}$$

$$I_{c4} = (I_0 + I_{in})e^{\frac{V_{in}-V_o}{V_t}} \quad \text{Eq. 5.21}$$

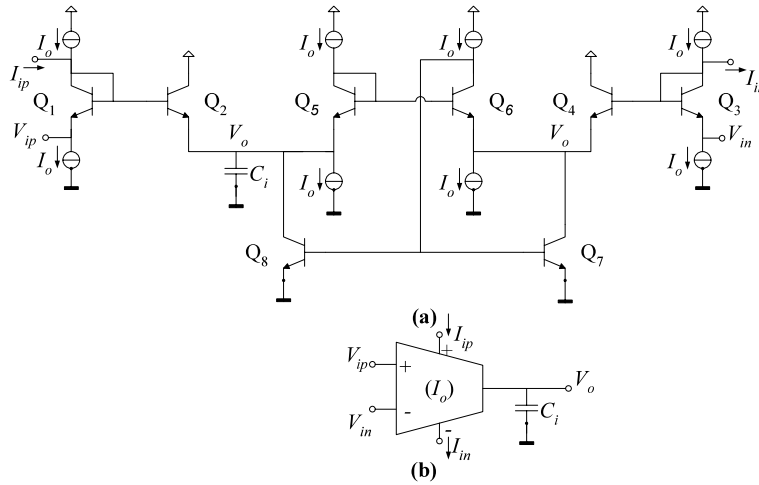


Figura 5.8 Integrador *log domain* classe AB

Com o diferencial das correntes realizado por Q7 e Q8 podemos escrever a lei das correntes de *Kirchhoff* no circuito como:

$$C \frac{dV_o}{dt} = (I_0 + I_{ip})e^{\frac{V_{ip}-V_o}{V_t}} - (I_0 + I_{in})e^{\frac{V_{in}-V_o}{V_t}} \quad \text{Eq. 5.22}$$

6 Circuitos Implementados

Seguindo o procedimento descrito na seção 5 e observando as necessidades dos algoritmos desenvolvidos, realizamos a implementação de alguns circuitos de forma a observar o comportamento dos algoritmos em meios menos ideais. A seguir descreveremos o procedimento para a implementação dos filtros gaussianos de primeira e segunda ordem, a realização de suas escalas e o procedimento de amostragem do sinal.

6.1 Aproximação dos filtros wavelet

A curva obtida com a resposta ao impulso de qualquer filtro wavelet é chamada “wavelet-mãe” para o trabalho realizado decidimos utilizar da primeira e segunda derivadas da gaussiana para realizar a amostragem desejada. O procedimento padrão utilizado foi a aproximação de Padè para encontrarmos a função de transferência da primeira derivada, vista na figura 6.1. juntamente com sua aproximação, e logo em seguida simplesmente derivamos a função encontrada por padè para obtermos a segunda derivada da gaussiana. A equação para a primeira gaussiana utilizada é dada pela equação.. Para transformar tal equação no domínio do tempo .

$$\psi(t) = (2t + 6)e^{-(t-3)^2} \quad \text{Eq. 6.1}$$

Para se transformar a equação da curva no tempo em uma função de transferencia realizável, encontramos sua transformada de laplace e da transformada encontrada realizamos a desejada aproximação por padé. Nas equações abaixo vemos tanto sua transformada de laplace quanto o polinomio de padé para a aproximação considerando um numerador de ordem 6 e um denominador de ordem 10.

$$\mathcal{L}(\psi(t)) = \frac{1}{2}\sqrt{\pi} \operatorname{erfc}\left(\frac{1}{2}s - 3\right) s e^{\frac{1}{4}s(-12+s)} - (e^{-9})^2 e^9 \quad \text{Eq. 6.2}$$

$H(s)$

$$= \frac{(9.63)10^{13}s^6 + (4.427)10^{17}s^5 - (3.431)10^{21}s^4 + (2.18)10^{24}s^3 - (4.716)10^{28}s^2 + (8.8)10^{31}s^1 - (3.681)10^{30}}{127.5s^{10} + (1.55)10^6s^9 + (9.44)10^9s^8 + (3.73)10^{13}s^7 + (1.05)10^{17}s^6 + (2.16)10^{26}s^5 + (3.3)10^{26}s^4 + (3.65)10^{26}s^3 + (2.8)10^{29}s^2 + (1.33)10^{32}s^1 - (2.98)10^{34}}$$

Como resultado da interpolação, na figura 6.1 comparamos a resposta ao impulso da função de transferência com a resposta no tempo desejada da primeira derivada da

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 62.22 \end{bmatrix} \quad C = [23.85 \quad 54.33 \quad -29.78 \quad 16.84 \quad -4.171 \quad 0.6576 \quad 0.2434 \quad 0 \quad 0 \quad 0]$$

Figura 6.2 Matrizes para representação em espaço de estados do filtro gauss1(primeira derivada da gaussiana).Elementos em escala de nA.

Na figura 6.3 vemos a resposta ao impulso do filtro desejado e sua resposta na frequencia.

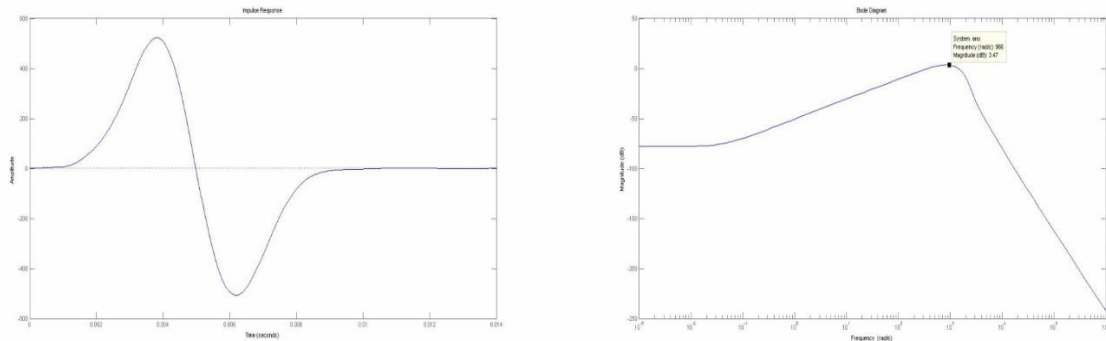


Figura 6.3 Resposta ao impulso do filtro orthonormal e resposta na frequencia.

Para gerar a segunda derivada da gaussiana apenas derivamos a função de transferência da primeira derivada multiplicando a mesma pelo operador derivativo S e em seguida realizamos o mesmo procedimento para encontrar as matrizes de espaço de estados. O resultado pode ser observado nas matrizes a seguir. Ao se olhar com cuidado vemos que a única diferença entre as matrizes geradas para a primeira e segunda derivadas está na matriz C . Tal fato é devido ao fato de ambas as funções de transferência apresentarem os mesmos coeficientes sendo a matriz C responsável por implementar os coeficientes derivativos necessários. Caso necessário poderíamos implementar duas derivadas a mais utilizando a mesma função de transferência e mantendo a causalidade do filtro. Tal prática pode gerar respostas menos estáveis uma vez que são feitas sobre a aproximação de padé e não sobre a função de transferência

original. Vemos na figura 6.5 as imagens da resposta ao impulso e resposta em frequencia do filtro implementado em espaço de estados.

$$A = \begin{bmatrix} 0 & 642.7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -642.7 & 0 & 895.5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -895.5 & 0 & 1080 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1080 & 0 & 1241 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1241 & 0 & 1426 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1426 & 0 & 1702 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1702 & 0 & 2185 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2185 & 0 & 3228 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3228 & 0 & 7070 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -7070 & -12160 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 62.22 \end{bmatrix} \quad C = [349.20 \quad 420. \quad 304.7 \quad -270 \quad 200 \quad -63.63 \quad 11.19 \quad 5.319 \quad 0 \quad 0]$$

Figura 6.4 Matrizes para representação em espaço de estados do filtro gauss2 (segunda derivada da gaussiana). Elementos em escala de nA

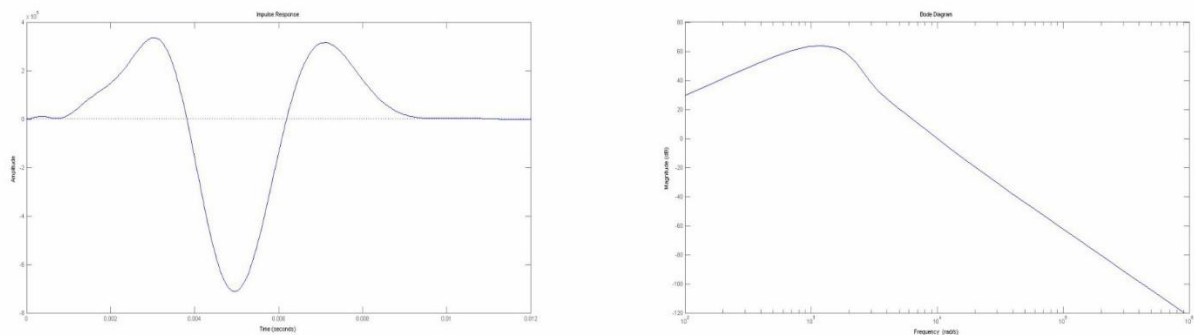


Figura 6.5 Resposta ao impulso do filtro orthonormal e resposta na frequencia.

6.2 Elaboração do filtro Wavelet

A topologia da filtro elaborado utiliza-se é a mesma da apresentada na figura 5.8. A diferença entre o integrador da figura 5.8 e o utilizado basicamente está no fato de utilizarmos transistores MOSFET para compor o integrador. Para operar utilizando o princípio de circuitos translineares polarizamos os transistores na região de inversão fraca. Transistores MOSFET em inversão fraca, para circuitos translineares, apresentam a mesma relação exponencial de corrente que os bipolares e para operar de tal forma devem apresentar uma tensão entre a porta e a fonte significativamente menor que sua tensão de threshold e uma tensão entre o dreno e a fonte superior a 100mV(4 vezes a tensão térmica V_t). Vemos a equação 6.3 que descreve a corrente no transistor MOSFET em inversão fraca.

$$I_d = I_{d0} \left(\frac{W}{L} \right) e^{\frac{V_{gs}}{nV_t}} \quad \text{Eq. 6.3}$$

Para implementar a célula do integrador da figura 6, escolhemos a tecnologia *TSMC 0.18μm*. Todas as células implementadas neste trabalho foram realizadas por meio do software *CADENCE*. O buffer de tensão formado por Q5 e Q6, e o espelho formado por Q7 e Q8, formam juntos um espelho de corrente low-voltage. Para que estas estruturas funcionem adequadamente, as seguintes condições devem ser satisfeitas:

- $V_{TH6} \geq V_{DS7}$
- $V_{GS5} \leq 2V_{TH6}$

As medidas dos transistores Q1, Q2, Q3, Q4, que formam os integradores log-domain escolhidas que atendem aos requisitos para inversão fraca foram:

$$W = 100 \mu m$$

$$L = 1 \mu m$$

E para os transistores Q5, Q6, Q7 e Q8, as medidas escolhidas foram:

$$W = 1 \mu m$$

$$L = 20 \mu m$$

Vemos a realização da topologia na figura 6.6 e 6.7. Nessa implementação incluímos as fontes de corrente para polarização e um respectivo estágio de polarização das fontes de corrente propostas.

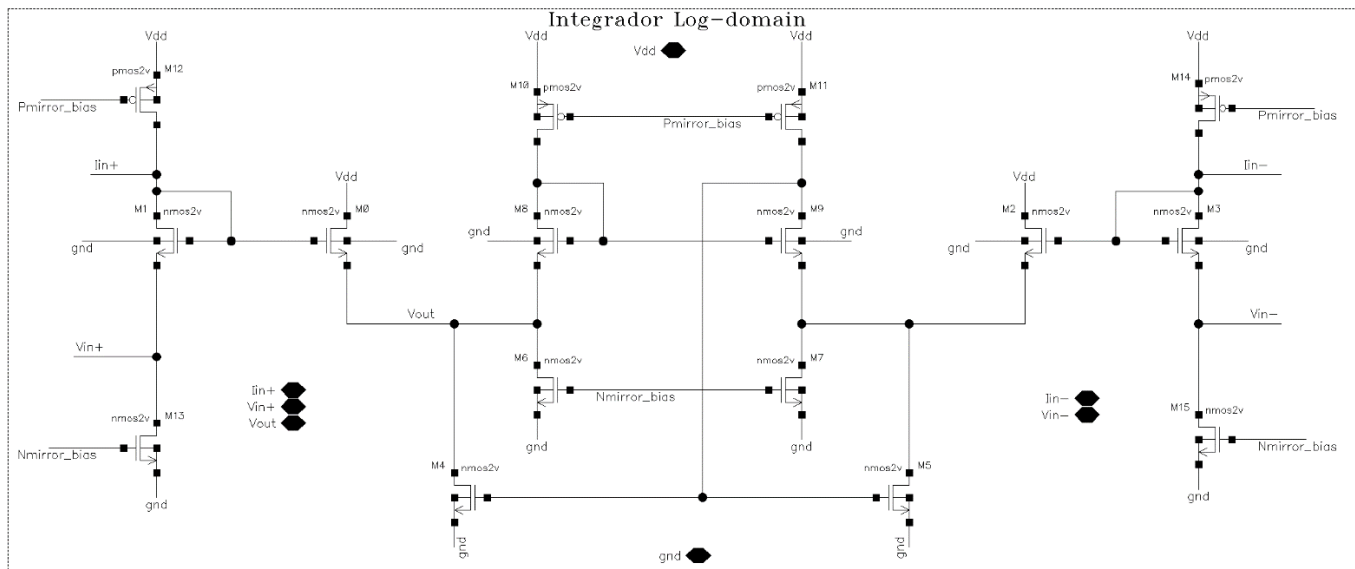


Figura 6.6 Integrador e fontes de corrente.

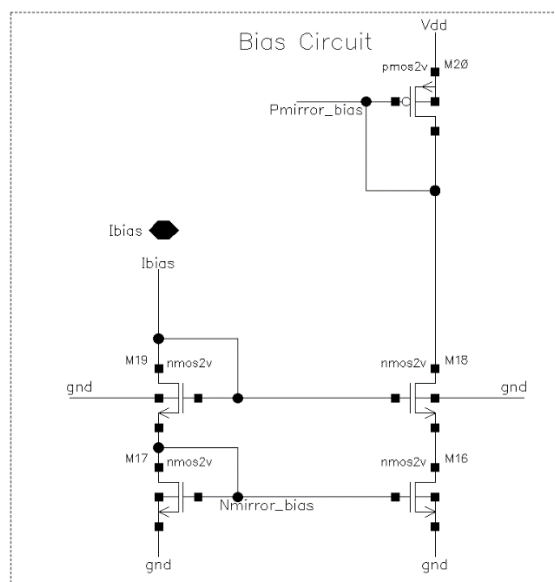


Figura 6.7 Estágio de polarização das fontes do integrador

Para encontrar o valor adequado de V_{ref} , foi realizado um teste simples, fixando V_{ip} e V_{in} em vários valores de V_{ref} , e conectando uma fonte de tensão em forma de rampa variando de 0 V a V_{DD} na saída ao invés de um capacitor. A tensão V_{DD} escolhida foi de 1.8 V. Mede-se então, o intervalo em que a tensão na fonte de Q6 é igual à tensão na fonte de Q5, ou seja, a faixa de operação para a qual a equação.. é válida. O teste foi realizado com fontes de corrente de 1.4uA e 300pA de forma a cobrir o máximo das correntes de polarização utilizadas. Com base no teste realizado decidimos polarizar todas as células com uma tensão de 200mV.

Após decidida a tensão de polarização a ser utilizada realizamos a conexão das células integradoras de forma a realizar o filtro em espaço de estados gerado. Nas matrizes obtidas, cada elemento representa a corrente de polarização de cada um dos integradores utilizados e indica a forma que cada célula deve alimentar e realimentar suas seguintes ou a saída de acordo com a equação de espaço de estados mostrada na equação 5., dado que a matriz D é nula, temos a seguinte implementação do filtro em circuito na figura 6.8.

Podemos ver também nas figuras 6.9 e 6.10 a resposta ao impulso e a resposta em frequência dos filtros desenhados e como se assemelham ao estimado no modelo computacional. Utilizamos uma função impulso de 1uA e largura de 100us.

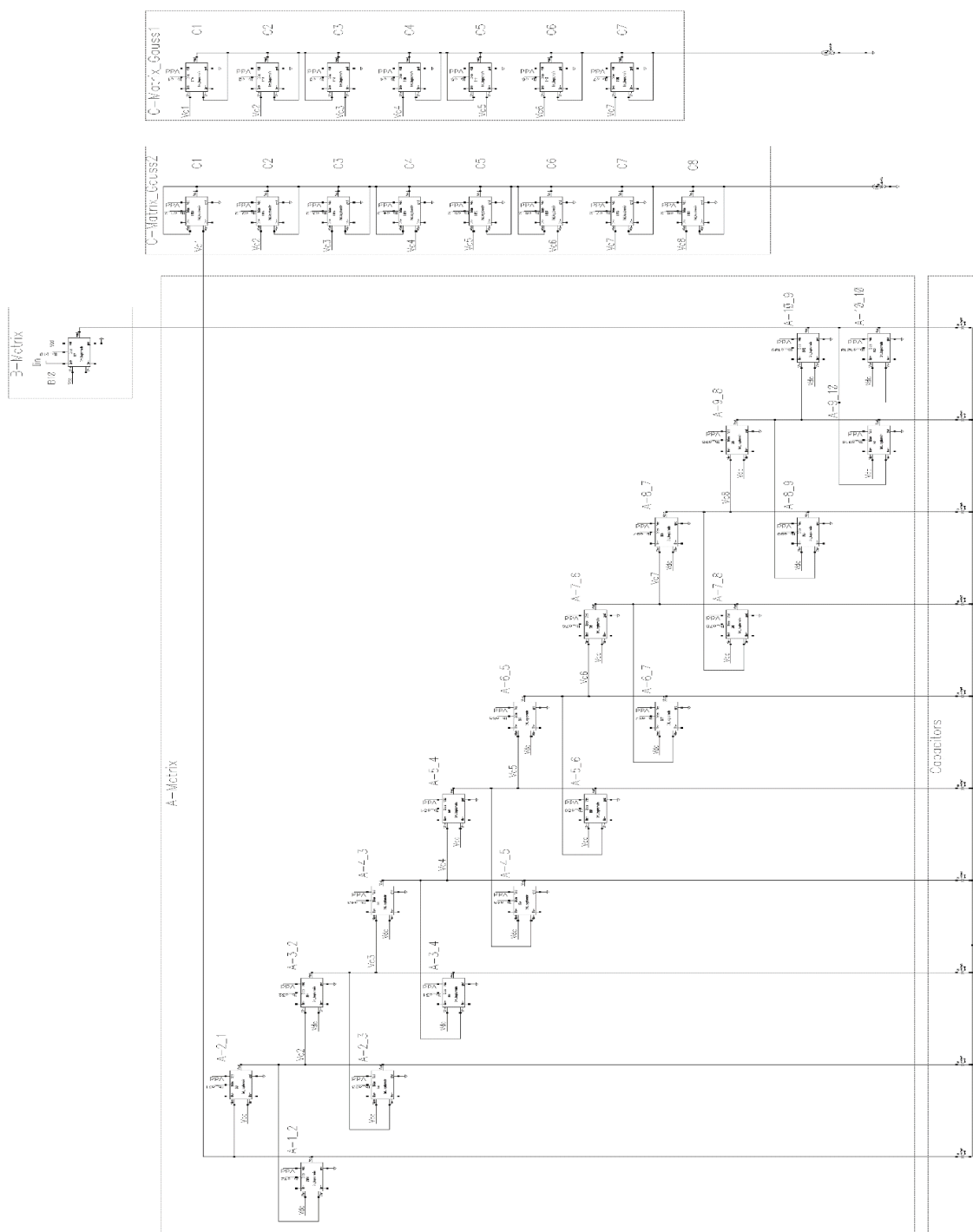


Figura 6.8- Esquemático dos filtros wavelet *gauss1* e *gauss2*

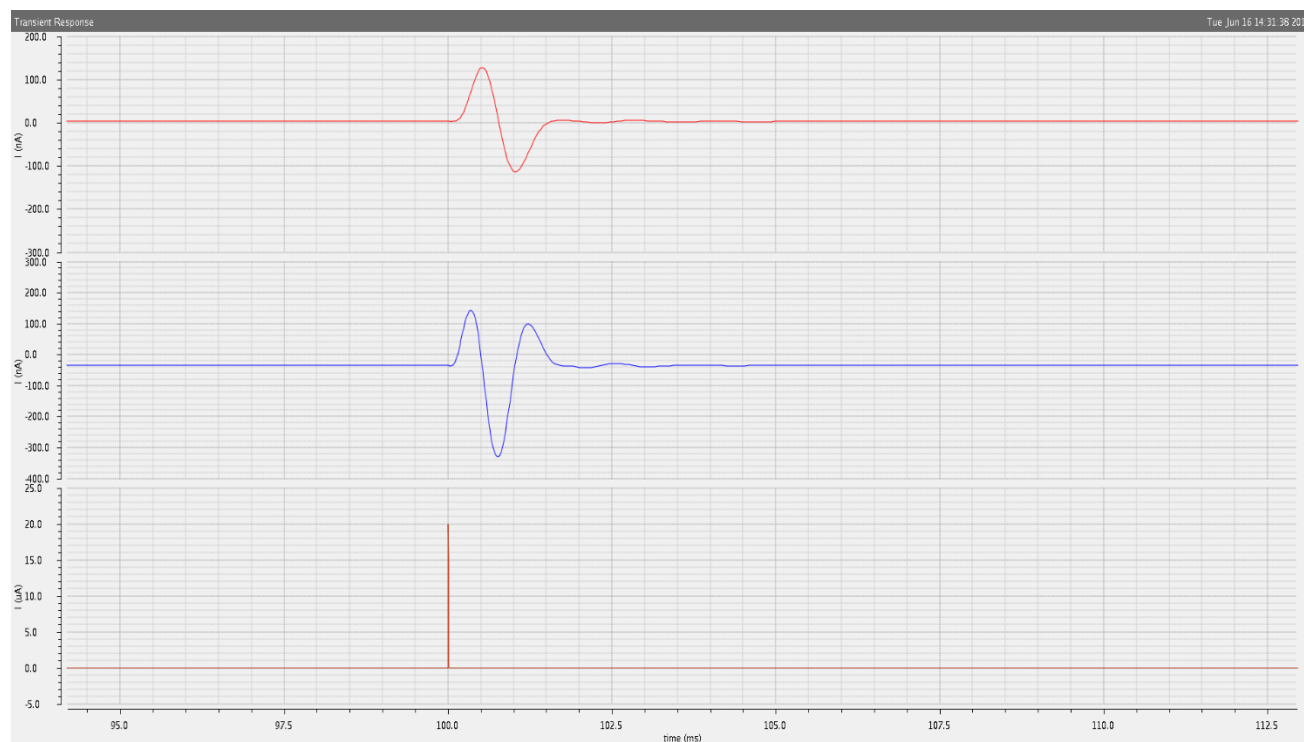


Figura 6.9 Resposta ao impulso do filtro wavelet

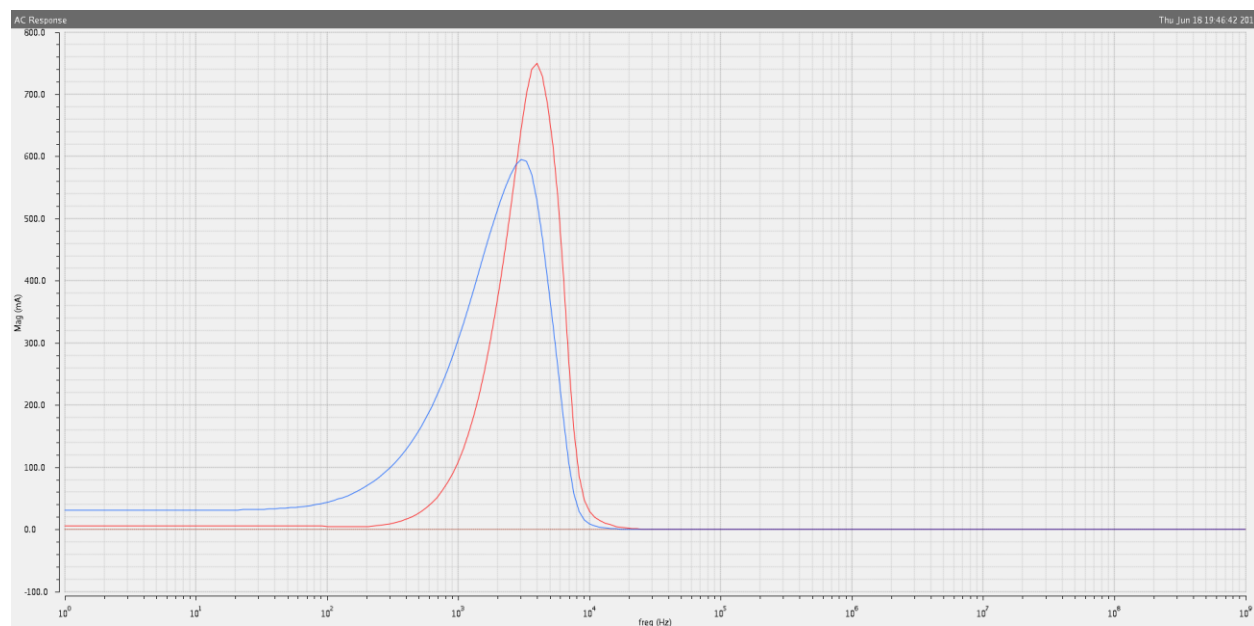


Figura 6.10 Resposta em frequência do filtro wavelet

6.3 Circuito para amostragem

Para a amostragem do sinal desejado elaboramos tres circuitos específicos, um comparador em modo corrente, um detector de borda e um *sample and hold*. Vemos cada um dos circuitos nas figuras 6.12 a 6.14.

A topologia do comparador utilizada foi inspirada no circuito publicado por (Neeraj K. Chasta, 2012). Nesta topologia, vemos três estruturas básicas, um pré-amplificador, um *latch* e um estágio de saída inversor. O pré-amplificador converte a corrente de entrada em um nível de tensão pelo espelho de corrente. Esse estágio também previne o comparador de ser afetado pelo *kickback* do *latch*. O estágio de *latch* é uma estrutura com feedback positivo que determina qual dos ramos do comparador apresenta maior valor e transiciona de forma abrupta sua saída. Tal transição é repentina devido ao *feedback* positivo e pode ser refletida na entrada pelo efeito acima citado, chamado *kickback*. O estágio de saída inversor tem como objetivo saturar o valor do latch de forma a alcançar os níveis digitais de VDD e 0 volts.

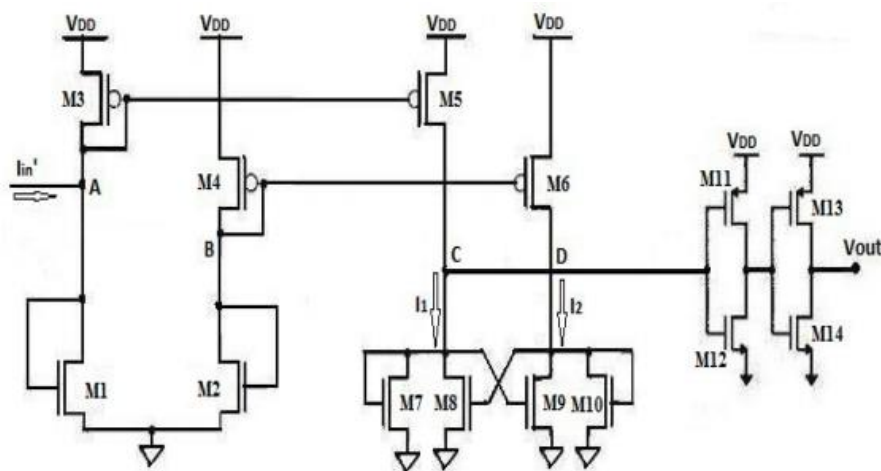


Figura 6.13 Comparador em modo corrente.

O circuito detector de borda funciona juntamente com o *sample and hold* modo a gerar uma amostra todos os momentos que o comparador transiciona. O circuito detector de borda é apenas uma combinação de portas lógicas enquanto o *sample and hold* funciona com duas chaves de modo a carregar e descarregar um capacitor com os valores de tensão desejados. Como desejamos utilizar duas fontes para decisão de pontos a serem amostrados, todo o circuito é replicado, tendo uma cópia a função de

amostrar os pontos de máximo e mínimo e a outra a função de amostrar os pontos de inflexão.

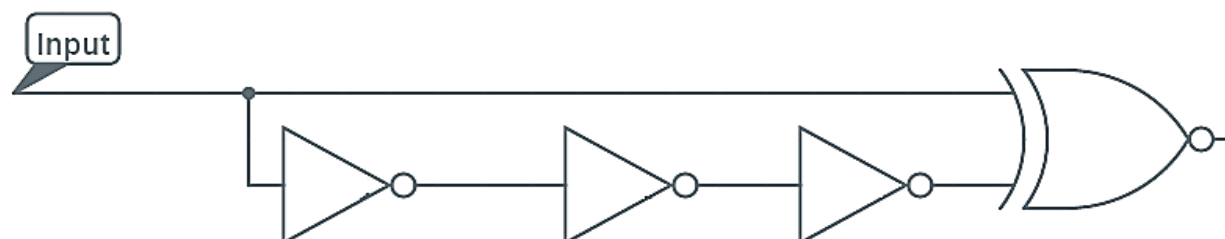


Figura 6.13 Circuito de detecção de borda

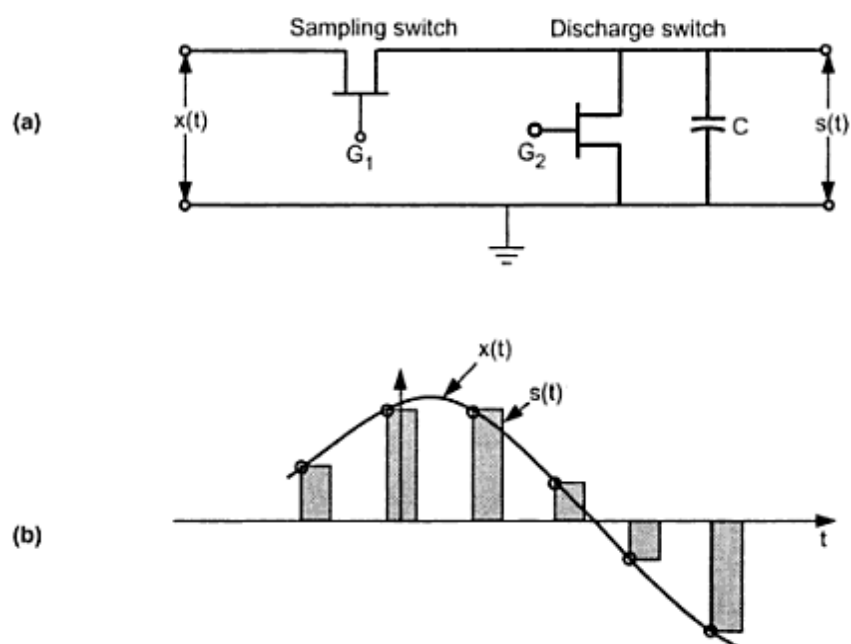


Figura 6.14 circuito para *sample/hold*. Vemos em (a) o esquemático elaborado e em (b) o comportamento esperado.

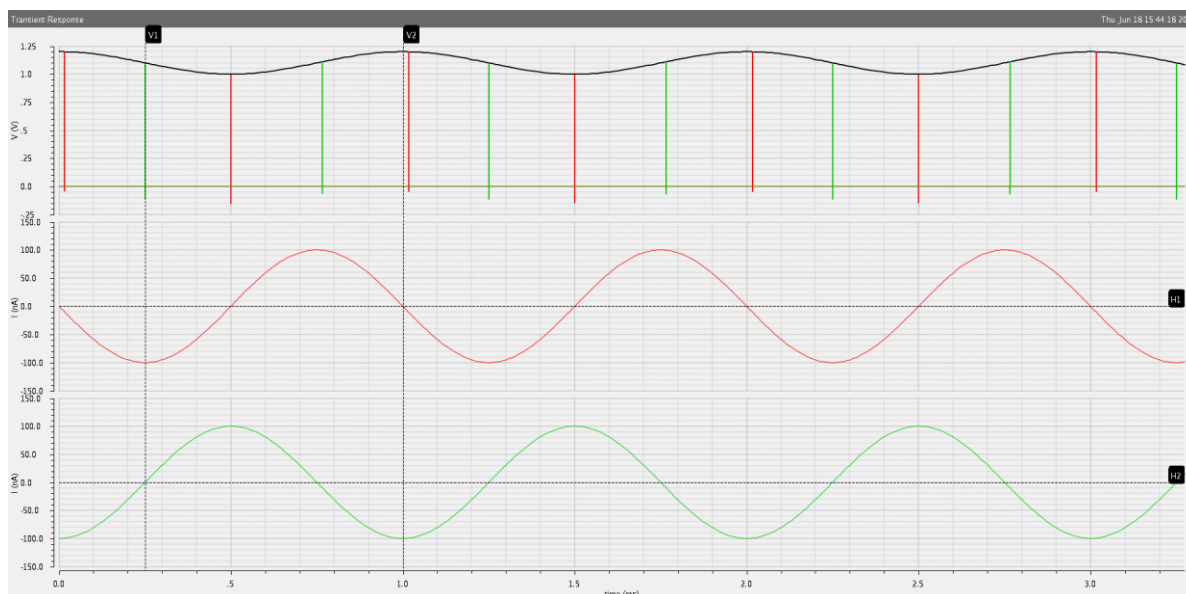


Figura 6.15 Resultado do circuito de amostragem. Vemos em vermelho e verde as fontes de corrente para a decisão de amostragem do sinal. Em preto vemos o sinal que desejamos amostrar os pontos amostrados pelas respectivas fontes de corrente.

Podemos ver o esquemático do circuito para *sample/hold* elaborado na figura 6.14, com a diferença de termos utilizado transistores NMOS e PMOS para cada chave a fim de melhorar a amostragem. O resultado do circuito de amostragem é visto na figura 6.15. Utilizamos duas fontes de corrente defasadas de 90 graus entre si para determinar os pontos de máximo mínimo e inflexão e uma fonte de tensão a ser amostrada. Cada fonte de corrente tem o papel de representar cada uma das transformadas wavelet utilizadas.

7 Resultados e performance

Utilizando o material apresentado até esse momento, foram criados algoritmos em *matlab R2013a*[®] para um estudo de viabilidade do método. Os algoritmos criados têm como objetivo gerar os pontos da conversão e extrair o expoente de *Lipschitz* dos pontos necessários, interpolar os pontos mostrando a possível recuperação do sinal a partir da amostragem feita pelo conversor, e extrair importantes figuras de mérito para se comparar com conversores comuns.

7.1 O Conversor

Seguindo o diagrama de blocos da figura 3.5 foi montado um algoritmo para extrair os dados necessários. Um sinal qualquer é gerado ao início do algoritmo e em seguida o mesmo passa por uma *Waveletgaus1* e *gaus2* gerando suas transformadas. A transformada *Wavelet* é implementada pela função *Continuous Wavelet transform* da *Wavelet toolbox*. É então observado o valor de saída do banco de filtros e no momento que a saída do filtro cruza o eixo das abscissas, é habilitada a amostragem do valor do sinal, que se deve referir ao valor da amplitude, como é possível ver na figura 7.1

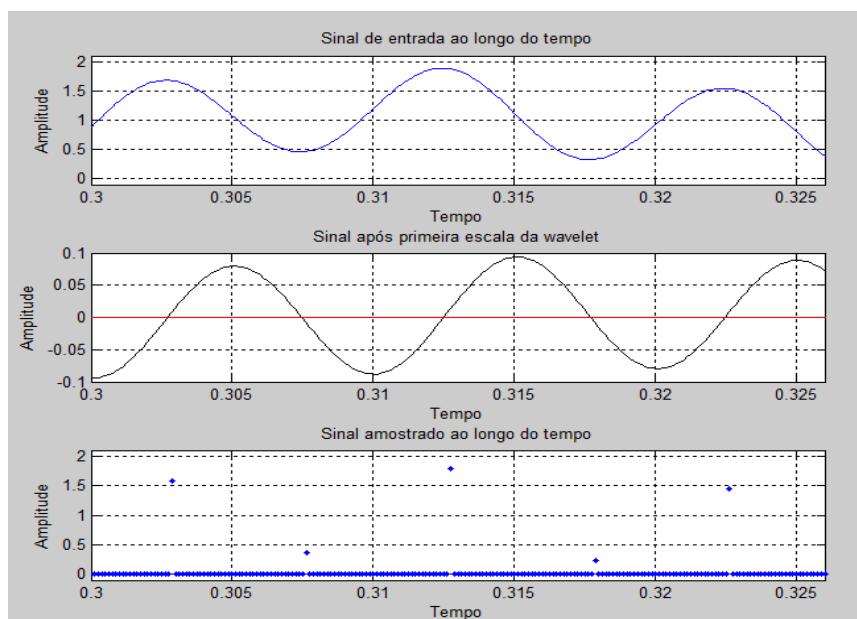


Figura 7.1 Processo de amostragem da amplitude

Ao mesmo tempo que a amplitude é amostrada, ocorre o processo paralelo para a amostragem do expoente de *Lipschitz* extraído do filtro de segunda ordem. Existe a necessidade de o filtro de segunda ordem estar sempre ativo devido à segunda função que o mesmo tem, fornecer a informação temporal da ocorrência de uma inflexão no sinal. A extração do *Lipschitz* é feita por uma média dinâmica ao longo das escalas dos filtros utilizados, conforme proposto em 3.3.1. O sistema para extração do *Lipschitz* e sinalização de inflexão é mostrado na figura 7.2.

Com os dados amostrados graças a habilitações por *zero-crossing* dos filtros *Wavelet*, nos resta realizar a conversão A/D dos mesmos. Neste caso devido à natureza do estudo, não foi feita uma completa conversão A/D, na prática o que foi realizado foi uma perfeita quantização do sinal em uma escala definida por 2^N , onde N é o número de bits desejado. A quantização é suficiente no caso, pois permite validar o conversor como um todo e nos permite a comparação com o que seriam conversores convencionais, também implementados como quantizadores perfeitos, seja variando a quantidade de bits tanto dos conversores tradicionais quanto dos conversores usados na nova topologia.

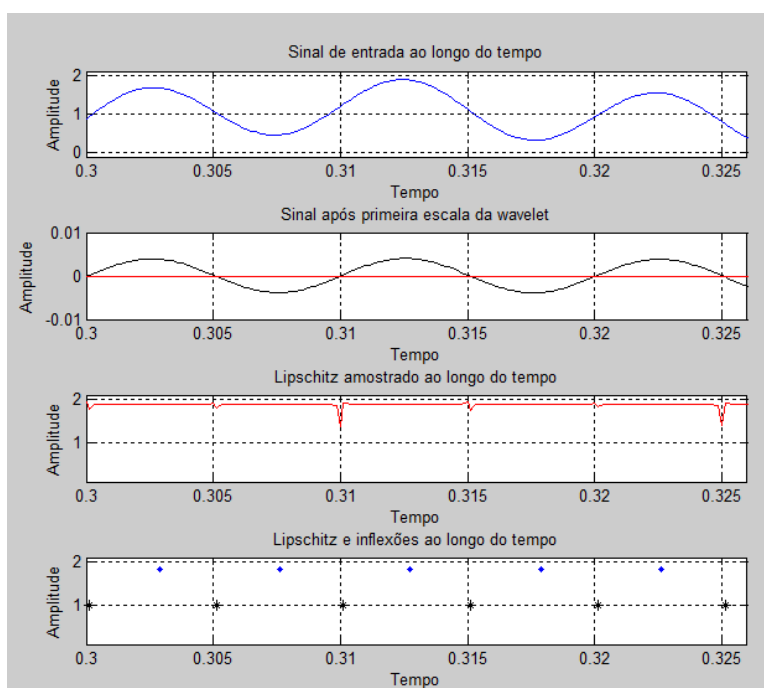


Figura 7.2 Extração do expoente de *Lipschitz* e localização das inflexões

7.2 O sinal reconstituído

A reconstituição do sinal é feita como descrito em 4.1. São determinados três pontos, sendo dois extremos definidos por amplitudes do sinal, gerando um máximo e um mínimo local, e o ponto de conformação entre as amplitudes definido pelo ponto de inflexão.

Apesar da aproximação consideravelmente boa, o algoritmo de reconstrução ainda precisa ser trabalhado. Na figura 7.3 tenta-se reconstruir um sinal dado pela Eq.(7.1). Apesar de em grande parte do tempo o sinal ser corretamente reconstruído, existem momentos, como no ponto 0.35 segundos, onde a interpolação ocorre de maneira imprecisa com relação ao sinal de entrada. Tais pontos não são amostrados pelos filtros utilizados e portanto necessitam ser tratados pelo algoritmo para interpolação. Nos resultados apresentados neste trabalho interpolamos pontos como este pela média dos seus vizinhos.

$$s(t) = 0.4 \sin(50 * 2\pi t) + 0.6 \sin(70 * 2\pi t) + 1.5 \quad \text{Eq. 7.1}$$

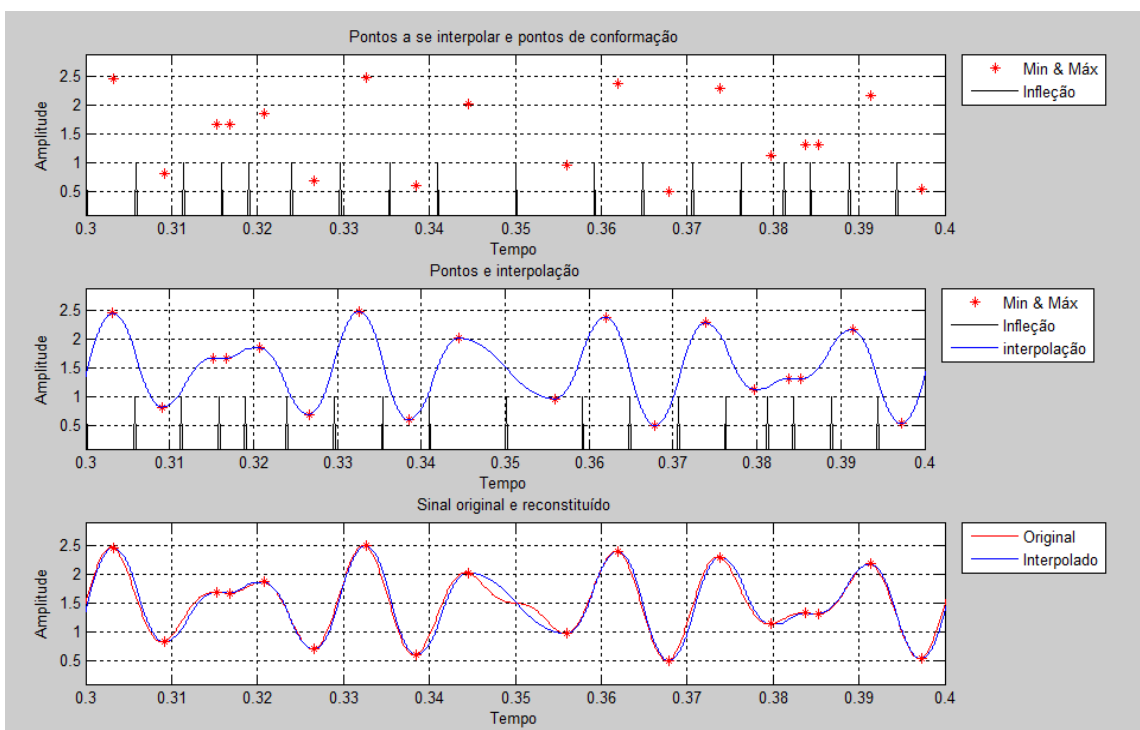


Figura 7.3 Reconstituição do sinal $s(t)$, pelo método polinomial

7.3 Comparação com conversores convencionais

Para validar o conversor baseado em wavelets utilizaremos na análise de figuras de mérito como o erro médio quadrático (E_{rms}) para todo sinal observado e o erro absoluto de conversão. Utilizaremos diferentes sinais diferentes para observar o desempenho do conversor em situações diversas. Faremos também uma comparação do espectro de frequência dos sinais que passaram tanto por conversores tradicionais quanto pelo conversor proposto nesse trabalho. Tal comparação tem como objetivo analisar os prós e contras analisando figuras de mérito tradicionais como *SNR*, *ENOB*, *THD*, *SNDR*, e erros de conversão, e apenas realizando essa análise podemos inferir situações onde o conversor proposto supera os métodos tradicionais e onde o mesmo não surge como uma boa solução.

7.3.1 Sinais e Figuras de mérito

Para a validação de desempenho do conversor checamos a possibilidade da conversão de sinais que apresentassem as problemáticas apresentadas. Com o objetivo de observar a capacidade de conversão de sinais reais e testar sua viabilidade, escolhemos sinais de eletrocardiograma e Imagens para realizarmos a conversão.

Com relação a análise do espectro de frequência, para uma comparação justa, utilizamos um sinal senoidal puro de fácil análise de sua transformada de fourier, sem harmônicos, sem ruídos e deslocados do eixo das abscissas de forma a não assumir valores negativos possibilitando a conversão pelo conversor wavelet. O sinal escolhido para tal comparação é mostrado na equação 7.2.

$$s(t) = \sin(50 * 2\pi t) + 2 \quad \text{Eq. 7.2}$$

7.3.2 Validação do desempenho para conversão de ECG

Na Figura pode-se ver o sinal de ECG escolhido para se realizar a conversão. O sinal foi obtido no banco de dados do instituto de tecnologia de Dublin. Para avaliar o desempenho para a conversão de sinais de eletrocardiograma utilizamos as figuras de

Erro relativo e Erro médio quadrático. Enquanto o Erro médio quadrático nos fornece um erro médio para todo o sinal analisado o Erro relativo nos mostra o pontos de máximo e mínimo de erros na conversão. As wavelets utilizadas para a conversão foram as Bases Biortogonais devido sua capacidade de captar pontos de máximo e mínimo mesmo em altas frequências.

Na figura 7.4 podemos ver como se dá a conversão para 4, 8 e 12 bits utilizando o método proposto respectivamente. Vemos que a partir de 8 bits temos uma conversão consideravelmente fiel. Lembrando que a amostragem neste caso não se dá apenas em pontos claros de máximo e mínimo, uma vez que o sinal está corrompido de ruído e o mesmo influencia adicionando diversos pontos de amostragem ao algoritmo. É possível verque a conversão não é exata e que a precisão é muito relativa a alguns eventos, vemos regiões com uma boa precisão e outras regiões não tao boas. O mesmo é possível de se ver na Figura 7.5 sobre o erro relativo de reconstrução, ao mesmo tempo que tempo regiões de erros muito baixos, inferiores a 1%, a grandes regiões de erros superiores a 30%.

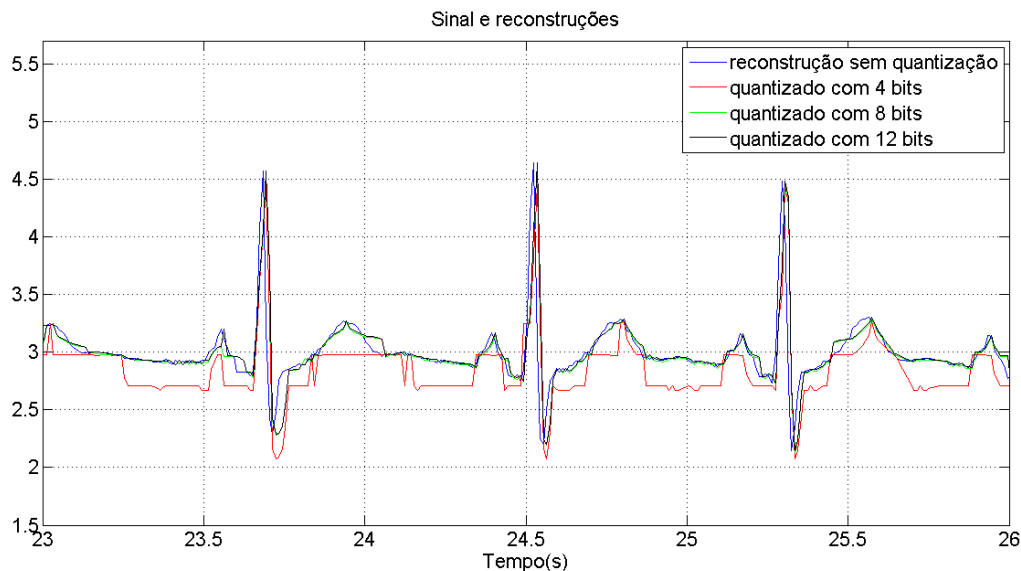


Figura 7.4 Sinal de ECG e reconstrução com diferentes quantidades de bits

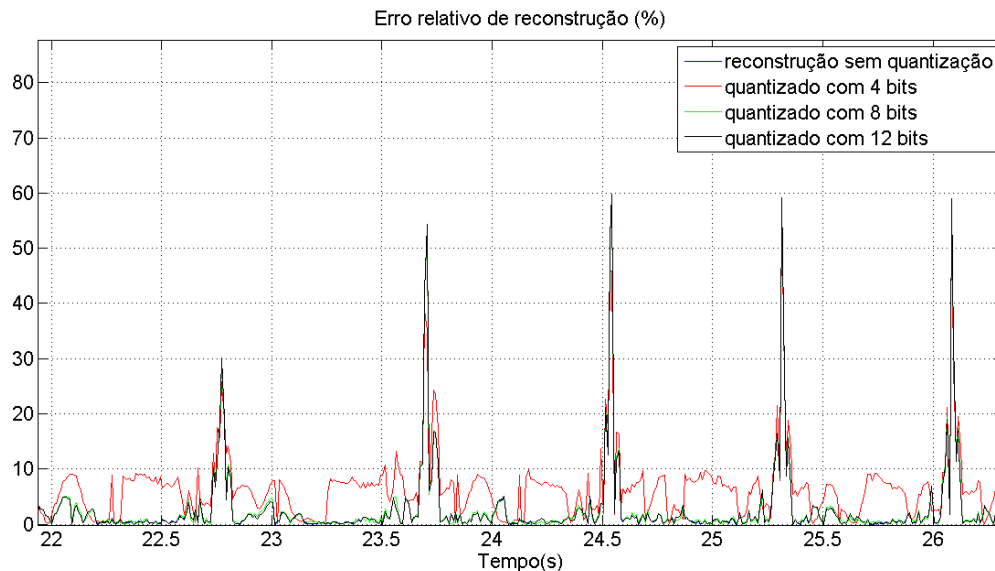


Figura 7.5 Erro relativo de reconstrução

Importante observar sobre a figura de erro relativo que parte do erro observado é devido a atrasos de fase inerentes dos filtros utilizados, como em praticamente todos os picos do ECG. Tais picos não são de grande prejuízo para o conversor, uma vez que a informação recuperada ainda é coerente com a original. Em outros pontos é possível observar uma clara deficiência do algoritmo de recuperar os resultados originais, gerando pequenas regiões de erros relativos altos (~5%). Tais problemas são mais preocupantes e demandam melhorias nos algoritmos de reconstituição.

Apesar de tais problemas, o erro relativo médio se manteve significativamente baixo para o sinal observado, conforme vemos na tabela 7.1.

$E_{rms}(\%)$	4 bits	8 bits	12 bits
<i>ECG</i>	0.2756	0.2091	0.2083

Tabela 7.1 Erro relativo médio para diferentes precisões

7.3.3 Validação do desempenho para conversão de imagens

Para a conversão de imagens, escolhemos duas imagens de 512x512 bits no formato bmp livre de compressão *Lena* e *Baboon* foram utilizadas pois apresentam características bem distintas e possibilitam uma melhor comparação do desempenho. O par de bases utilizado foram as biortogonais devido sua melhor capacidade de captar transições em alta frequência. Podemos ver ambas imagens originais na Figura 7.6. Para a conversão, transformamos cada imagem em um vetor concatenando todas as linhas e após todo o processo de conversão retornamos o vetor a suas dimensões originais na imagem. Dessa forma foi possível utilizar o mesmo algoritmo proposto inicialmente para operar sobre sinais unidimensionais. Decidimos validar o desempenho do algoritmo em cima de imagens de forma a checarmos se visualmente o resultado se assemelha com a imagem original, e como o erro de conversão se relaciona com a qualidade da reconstrução.



Figura 7.6 Baboon e lena

Observamos que enquanto a imagem *baboon* possui diversos pontos de transição de alta frequência, *Lena* apenas possui transições de alta nos contornos e na região das penas. Tal diferença é essencial para avaliar o desempenho do algoritmo de conversão em ambas as imagens.

Para a conversão utilizamos uma quantidade fixa de bits para o lipschitz e variamos a quantidade de bits para a amplitude dos pontos amostrados. Para cada caso

utilizamos 4, 8 e 12 bits. Observamos na Figura o resultado da conversão A/D e recuperação do sinal a partir do algoritmo do conversor D/A e na Tabela podemos ver o erro médio quadrático da imagem.

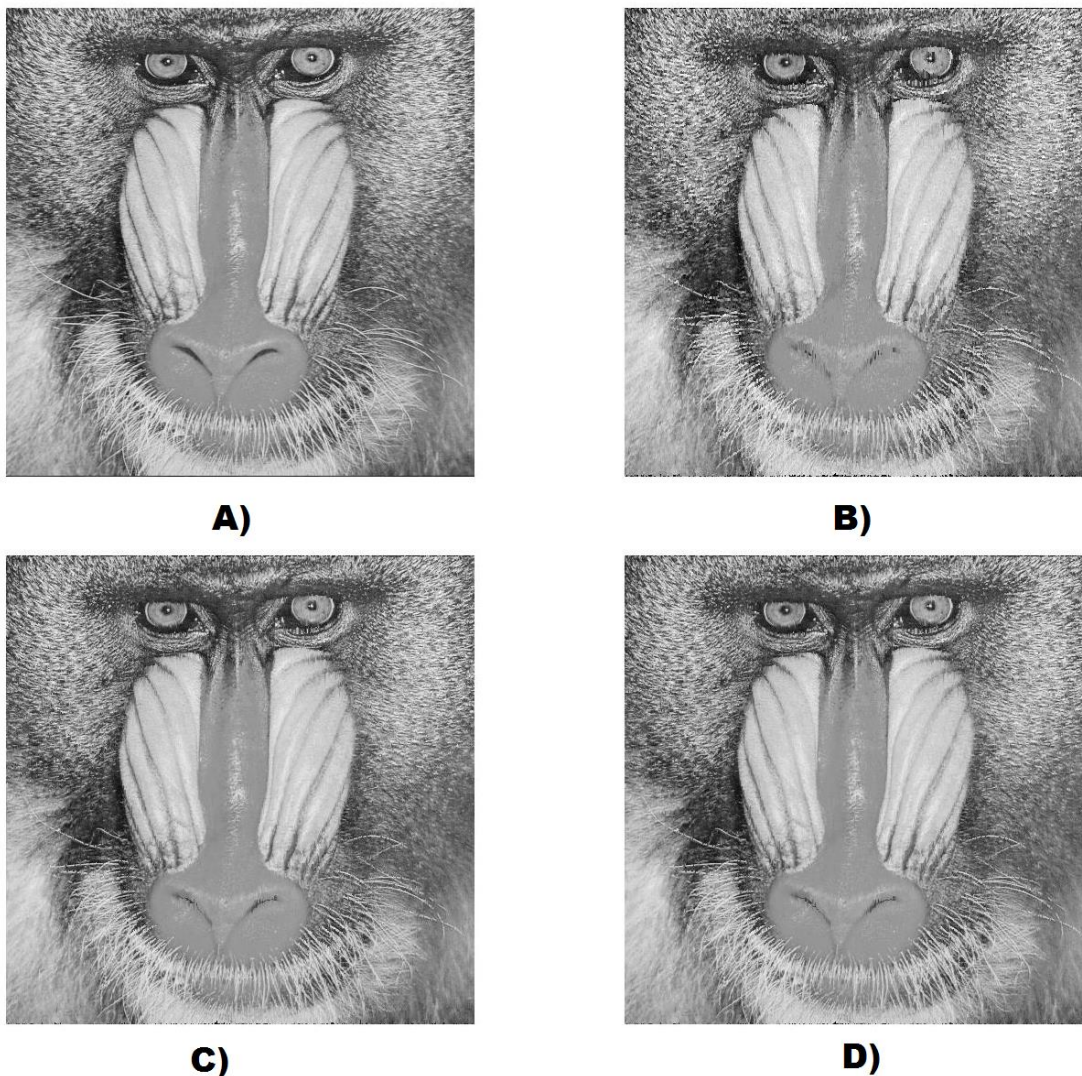


Figura 7.7 a) imagem original. b) conversão e reamostragem a partir de 4 bits. c) conversão e reamostragem a partir de 8 bits. d) conversão e reamostragem a partir de 12 bits

Pela imagem *Baboon* vemos poucos erros claros na conversão, regiões como os olhos e próximas às narinas mostram pequenos traços de erros, contudo uma análise do erro médio quadrático mostra uma outra figura sobre a qualidade da conversão e que provavelmente 4 bits seria o suficiente para se reconstruir tal imagem. Vemos com mais

clareza como as falhas da conversão se apresentam observando a *Lena* na figura 7.8. Os erros próximos a regiões de transições abruptas, ou contornos, não são mascarados nesta imagem e vemos os pontos onde podemos melhorar o algoritmo de reconstrução. Contudo, o erro médio quadrático se mostra menor nesta imagem, o que nos mostra que apesar de quantificar o erro, não seja a melhor figura de mérito para avaliar imagens.

**A)****B)****C)****D)**

Figura 7.8 a) imagem original. b) conversão e reamostragem a partir de 4 bits. c) conversão e reamostragem a partir de 8 bits. d) conversão e reamostragem a partir de 12 bits

$E_{rms}(\%)$	4 bits	8 bits	12 bits
<i>Baboon</i>	0.11	0.11	0.11
<i>Lena</i>	0.065	0.044	0.043

Tabela 7.2 Erro relativo médio para diferentes precisões

7.3.4 Validação do desempenho na frequência

Uma figura de mérito de grande importância para medir a qualidade de um conversor se dá na extração da relação sinal ruído(SNR) e quantidade de bits efetiva(ENOB) do mesmo. Utilizamos um sinal senoidal puro de 50Hz conforme descrito anteriormente na Eq 7.2. Comparamos o desempenho do conversor através do levantamento do espectro de frequência e o comparamos com um quantizador ideal com as mesmas quantidades de bits. A comparação foi feita entre o conversor desenvolvido e quantizadores ideais com a mesma quantidade de bits trabalhando em regime de oversample de 12 vezes e de 180 vezes a frequência do sinal de entrada. Na tabela podemos ver os dados levantados durante a comparação e nas figuras vemos os espectros de cada caso.

	Wavelet			12*Fs			180*Fs		
	4 bits	8 bits	12 bits	4 bits	8 bits	12 bits	4 bits	8 bits	12 bits
$E_{rms}(\%)$	0.06	0.0164	0.016	0.105	0.104	0.104	0.0145	3.85e-3	3.85e-3
SNR	47.58dB	56.69dB	57.76dB	30.93dB	56.55dB	63.67dB	35.33dB	59.05dB	83.91dB
SNDR	33.22dB	56.43dB	57.53dB	30.09dB	54.53dB	63.39dB	32.19dB	55.79dB	79.4dB
-THD	33.38dB	68.89dB	70.44dB	37.63dB	58.83dB	75.41dB	35.07dB	58.55dB	81.91dB
ENOB	5.23 bits	9.08 bits	9.26 bits	4.71 bits	8.77 bits	10.2 bits	5.05 bits	8.97bits	12.9 bits

Tabela 7.3 Comparação a nível de algoritmo entre o conversor desenvolvido neste trabalho e quantizadores perfeitos trabalhando a diferentes taxas de amostragem

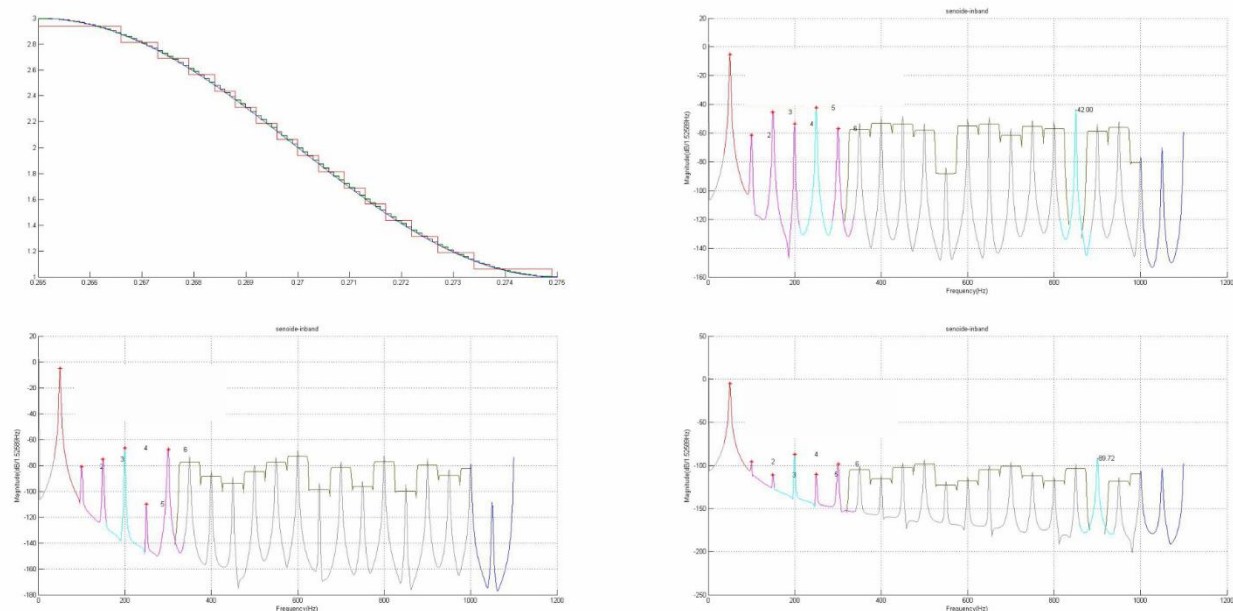


Figura 7.11 Conversão parcial por alta frequência de amostragem e o espectro de frequência para 4, 8 e 12 bits respectivamente.

Os resultados mostram a qualidade que o conversor nos fornece em especial na região de média resolução. Vemos que nas regiões de baixa e média resolução o conversor se mostra superior em praticamente todos os aspectos mesmo se comparado com o quantizador operando numa taxa de 180 vezes de oversample. Porém, muito devido a forma como a reconstrução é feita o conversor não parece progredir muito quando se eleva a resolução. Nesse aspecto podemos ver que o conversor D/A pode ser a restrição na melhoria do sistema.

7.4 Desempenho dos circuitos

Por meio do software cadence, procuramos validar os circuitos produzidos por meio de simulações. O objetivo principal é caracterizar o comportamento dos circuitos produzidos e verificar a viabilidade de cada um destes.

Durante o processo de validação dos circuitos, vários problemas inesperados apareceram, em especial problemas relacionados a fase dos sinais. Os filtros wavelet apresentavam atrasos de fase altos, e variantes com a frequência, inclusive essa variação se mostrou diferente quando comparamos diferentes escalas, o que tornou o

processo de calculo do coeficiente de lipschitz por meio de circuitos inviável até o momento. Apesar de tal característica, a diferença de 90 graus entre os filtros *gauss1* e *gauss2* se mostrou constante, o que possibilitou a amostragem dos sinais em questão, trabalhando a questão da diferença de fase.

7.4.1 Filtros Wavelet

Utilizando a topologia apresentada na seção 6.2, produzimos um bloco capaz de fornecer as primeiras duas derivadas gaussianas. Como já mostrado, realizamos testes para checar a resposta ao impulso e sua resposta em frequência sendo ambos os resultados satisfatórios. Em seguida variamos os valores das capacitâncias de cada um dos integradores para checarmos a variação do filtro com relação para cada escala desejada. Nas imagens 7.12 e 7.13 podemos ver o resultado para cargas de capacitores de 50pF, 100pF, 200pF, 400pF e 800pF de forma a obter cinco escalas das wavelets. Na figura 7.14 pode-se observar a variação da resposta em frequência de ambos os filtros com relação a escala. A medida que aumentamos a escala, deslocamos a frequência central do filtro, que por natureza tem a característica passa faixa, para frequências mais baixas.

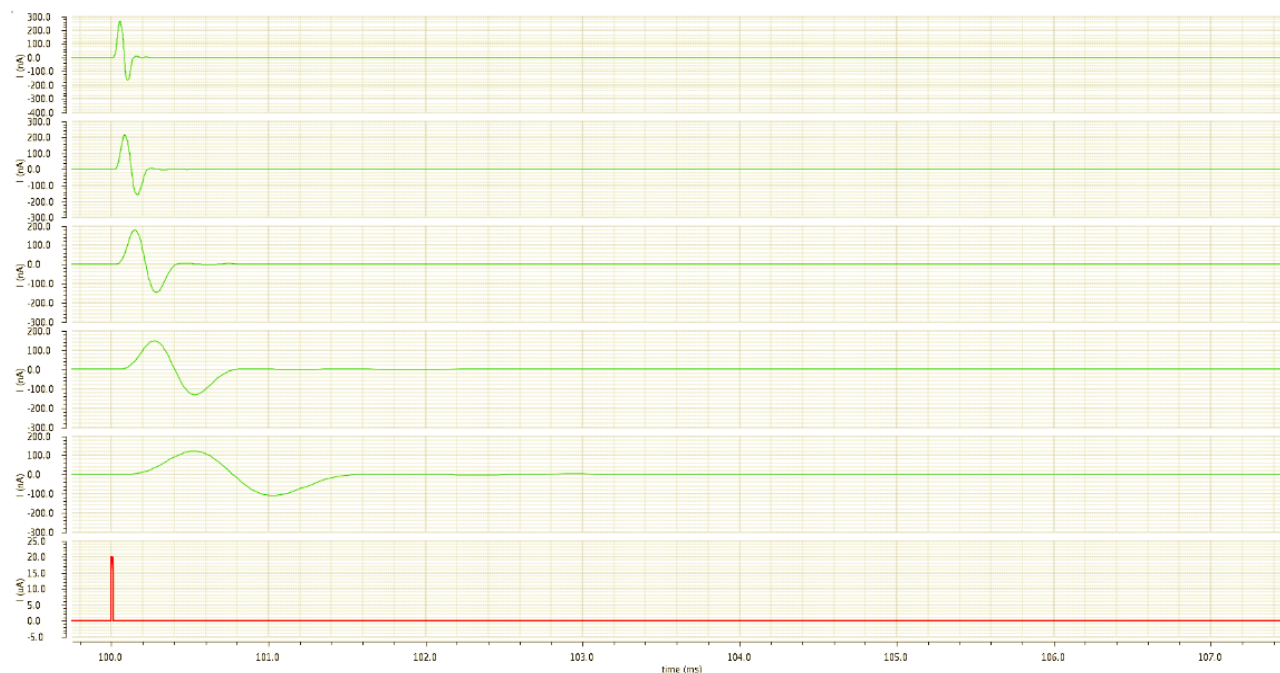
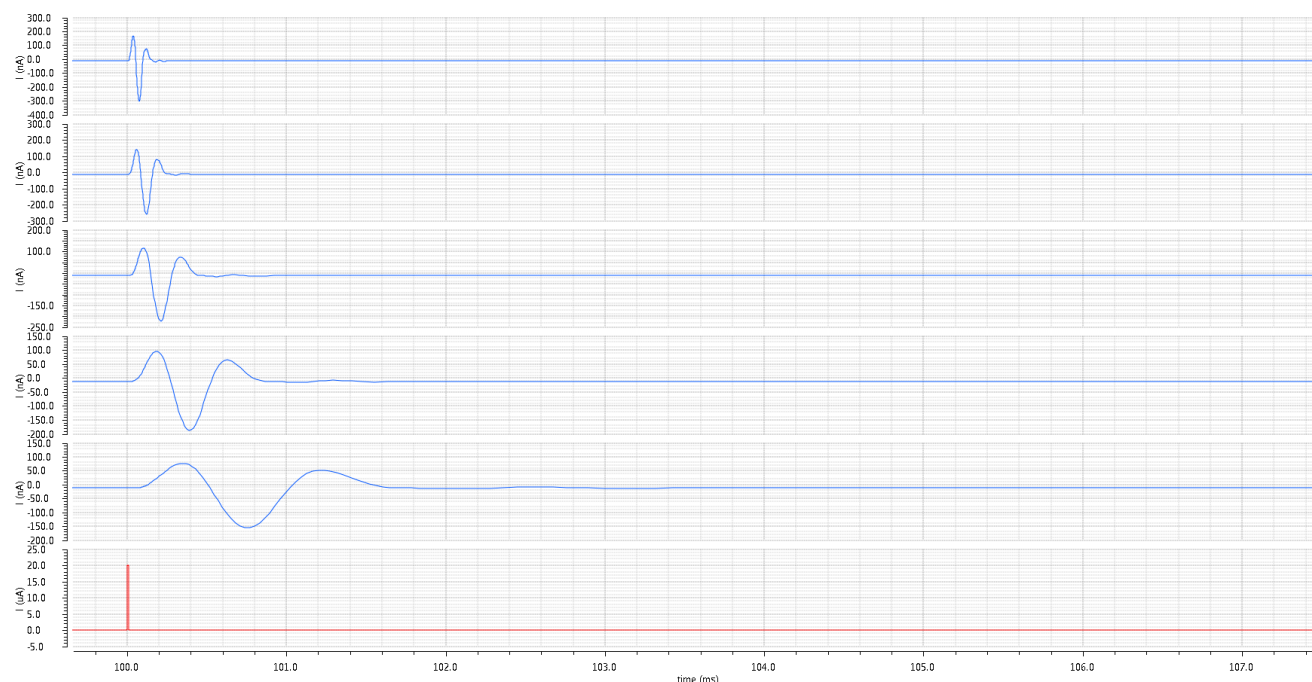


Figura 7.12 Resposta ao impulso do filtro gauss1

De forma a validar o comportamento de amostragem desejado para o funcionamento do conversor, alimentamos o filtro com uma família de ondas senoidais de forma que todo o sinal estivesse presente na banda do filtro wavelet. Podemos ver o resultado na Figura 7.15. O comportamento obtido esteve dentro do esperado, com a exceção de uma inesperada adição de fase que os filtros proporcionam ao sinal original comentada no início da sessão. Para a continuação do trabalho colocaremos os sinais utilizados em fase correta para realizar a validação. O *test bench* utilizado pode ser visto na figura 7.16.



7.13 Resposta ao impulso do filtro gauss2

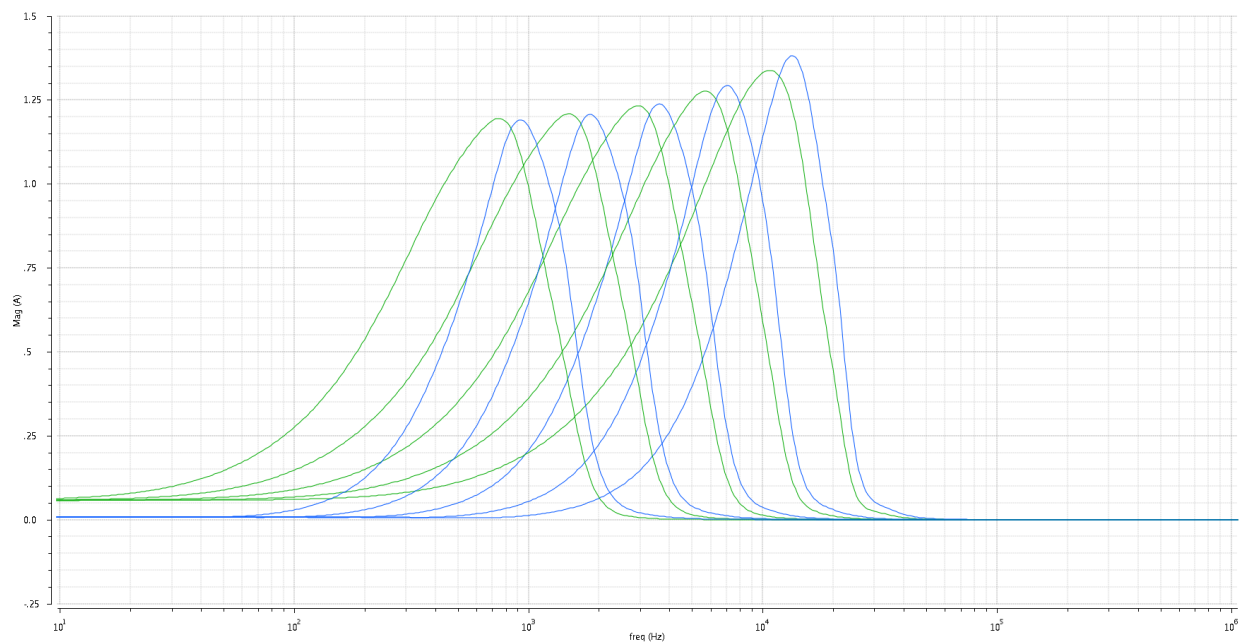


Figura 7.14 Resposta em frequência. Em verde as respostas do filtro gauss1 e em azul as respostas do filtro gauss2.

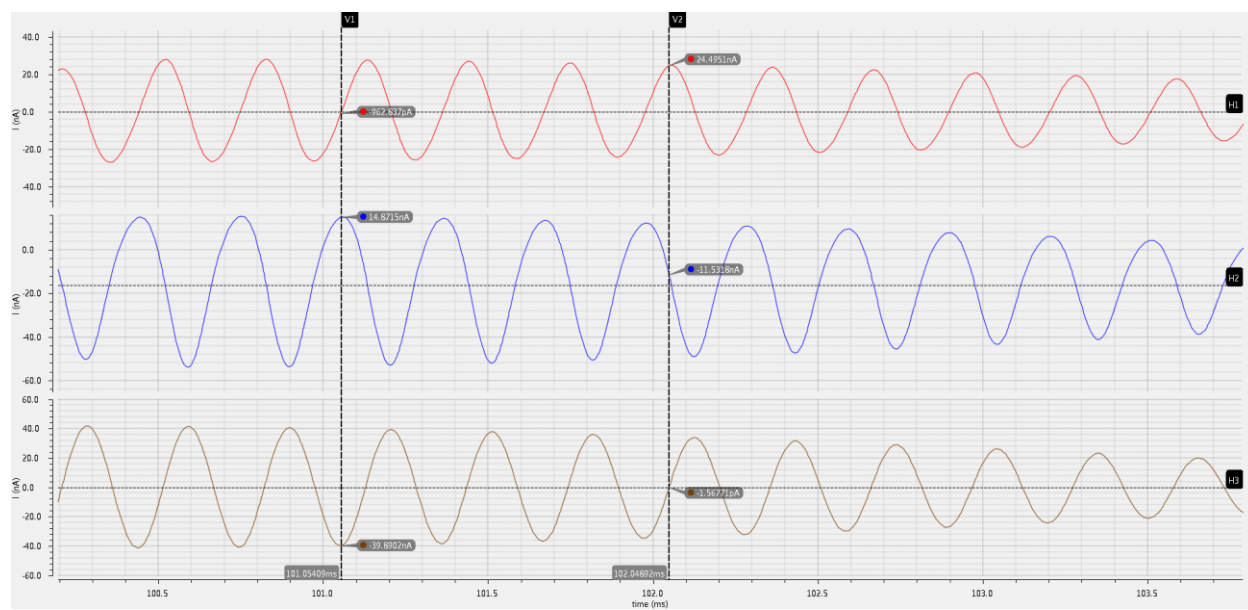


Figura 7.15 Detecção de máximos, mínimos e inflexões. Em marrom o sinal original, Em vermelho seu produto pelo filtro gauss1 e em azul seu produto pelo filtro gauss2

Com o resultado satisfatório dos filtros gaussianos, seguimos para a elaboração e testes dos circuitos restantes.

7.4.2 Circuitos de Amostragem

Vemos na Figura 7.16 os resultados do teste do circuito de amostragem elaborado, onde procuramos simular o comportamento do circuito para as entradas reais. No teste realizado alimentamos o circuito com duas fontes de corrente independentes defasadas de 90 graus e uma fonte de tensão para realizar a amostragem. O objetivo neste ponto era simular a entrada recebida pelo circuito elaborado para o filtro wavelet. Cada fonte de corrente representa uma transformada wavelet(gauss1 e gauss2) que possuem resposta em frequência parecida e uma adição de fase de 90 graus.

Observando os marcadores na figura 7.16 vemos que foi possível amostrar os pontos necessários para a reconstrução do sinal com uma precisão razoável.

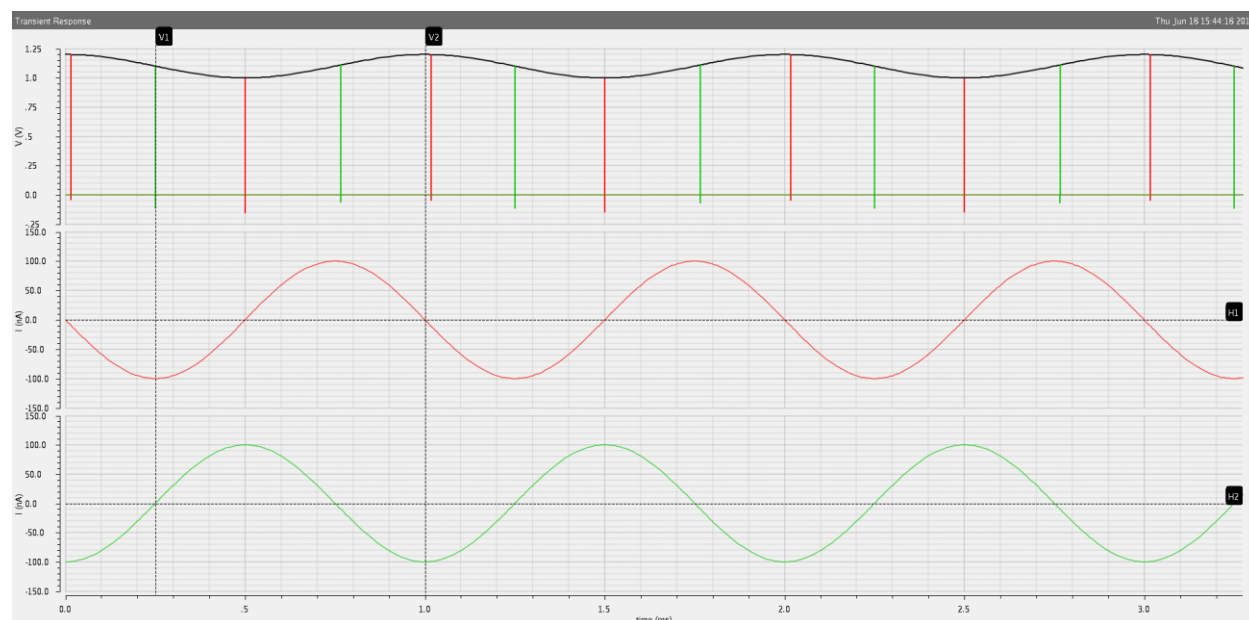


Figura 7.16 Simulação de funcionamento do amostrador. No primeiro gráfico vemos o sinal de interesse sendo amostrado nos pontos de máximo mínimo e inflexões. Nos outros dois em verde e vermelho vemos os sinais utilizados para a amostragem.

7.4.3 Cálculo do expoente de *Lipschitz*

Neste ponto utilizamos a calculadora do *CADENCE* para calcular o expoente de lipschitz de forma exata a partir das transformadas wavelet geradas. Foi montado um

test bench com dois filtros wavelet de escalas diferentes e um sinal senoidal cuja frequência estivesse dentro da banda dos filtros utilizados.

Na calculadora calculamos o módulo dos sinais e realizamos seus logaritmos de forma a satisfazer a expressão da equação

$$\alpha = \frac{\log_2 |WT(t, a_i)| - \log_2 |WT(t, a_{i-1})|}{\log_2 a_i - \log_2 a_{i-1}} - \frac{1}{2}$$

Vemos na figura 7.17 o resultado do calculo do expoente de lipschitz para um sinal senoidal. Vemos uma boa aproximação para os pontos de máximo, uma vez que esperamos um expoente de valor próximo a 2. Observamos no estanto que as duas escalas de wavelet utilizadas estavam fora de fase, tal fato juntamente com imperfeições das wavelets utilizadas podem prejudicar a aproximação desejada. Importante notar que o resultado, todavia, possui comportamento próximo ao teórico obtido por simulações, podendo mostrar que o resultado ainda é coerente com o esperado.

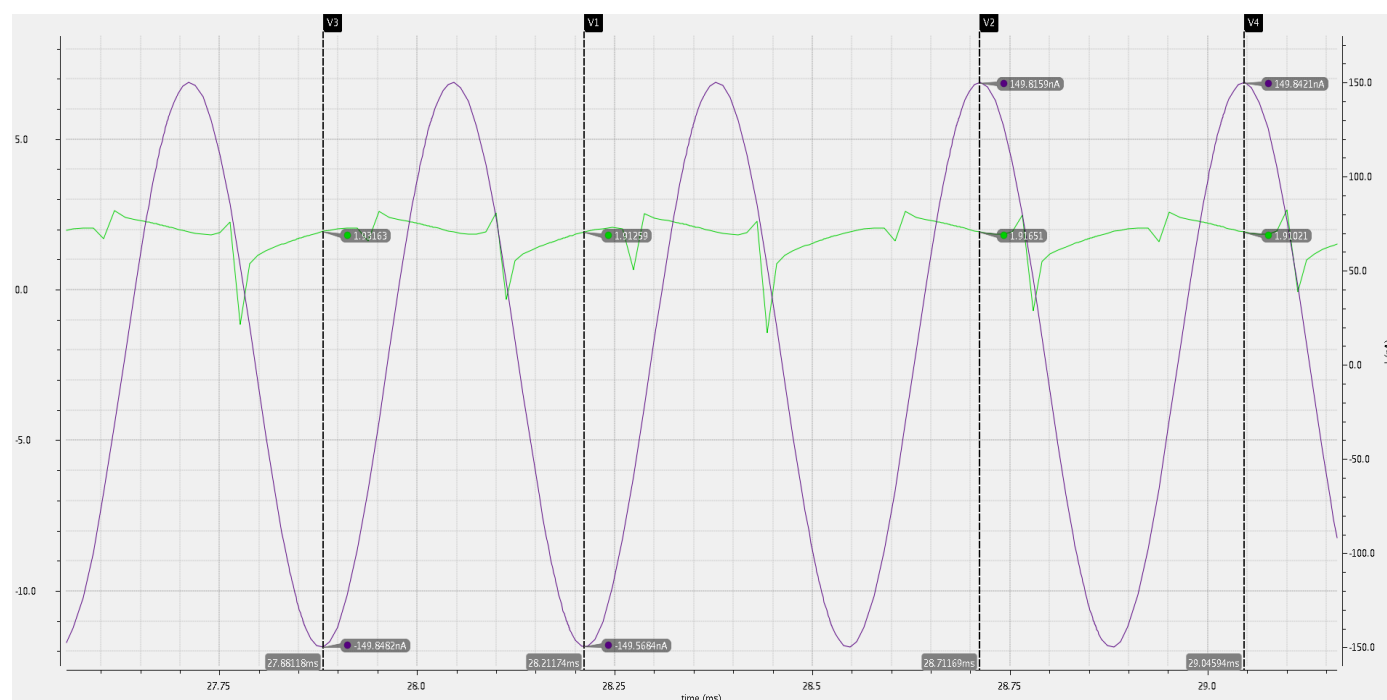


Figura 7.17 Sinal de interesse, roxo, e seu respectivo coeficiente de lipschitz, verde, calculado a partir da transformada gauss1

7.4.4 Amostragem a partir de transformadas Wavelet

Tendo completado os filtros wavelet e os circuito de amostragem, alimentamos os filtros com um sinal senoidal de frequência de 3.2K Hz e conectamos a saída dos mesmos ao amostrador por meio de um buffer de corrente. Houve a necessidade de se utilizar um buffer de corrente uma vez que o filtro wavelet não conseguia fornecer a corrente da forma necessária para o circuito de amostragem. Vemos na figura 7.18 o resultado da amostragem do sinal.

Observando a figura 7.18 vemos que foi possível amostrar os pontos de máximo mínimo e inflexões com uma precisão boa, porém vemos que há espaço para melhoria da precisão de amostragem.

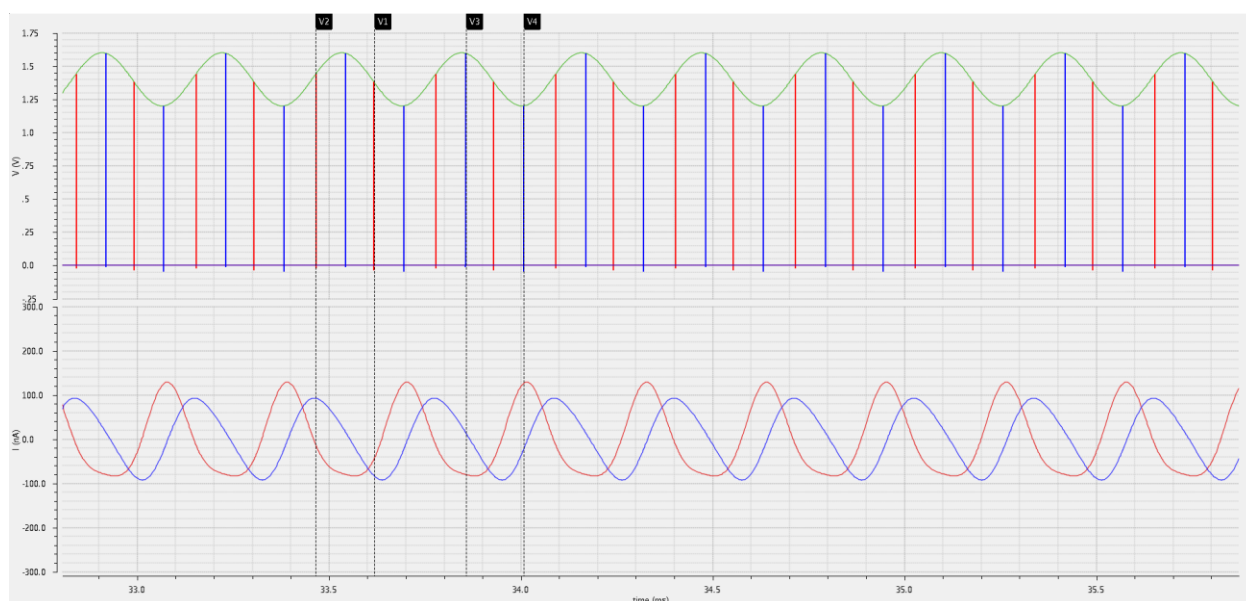


Figura 7.18 Circuito amostrador e transformadas wavelet utilizadas

7.4.5 Problemáticas encontradas

Conforme introduzido no início do tópico, encontramos diversos problemas relacionados a atraso de fase durante a elaboração do circuito. Conforme podemos ver na figura 7.19, os filtros utilizados possuem um atraso de fase variante ao longo de toda a banda do filtro. Nesta imagem vemos que a 1kHz o filtro *gauss1* apresenta um atraso de aproximadamente 153 graus enquanto a 10k Hz o atraso aumenta para mais de 600

graus. Tal variação de fase impossibilita o uso dos filtros da forma desejada ao longo da banda de interesse. Porém ao observarmos a diferença da fase entre os filtros *gauss1* e *gauss2* dentro desse mesmo intervalo, figura 7.20, vemos que o mesmos mantém o atraso sempre próximo aos 90 graus desejados.

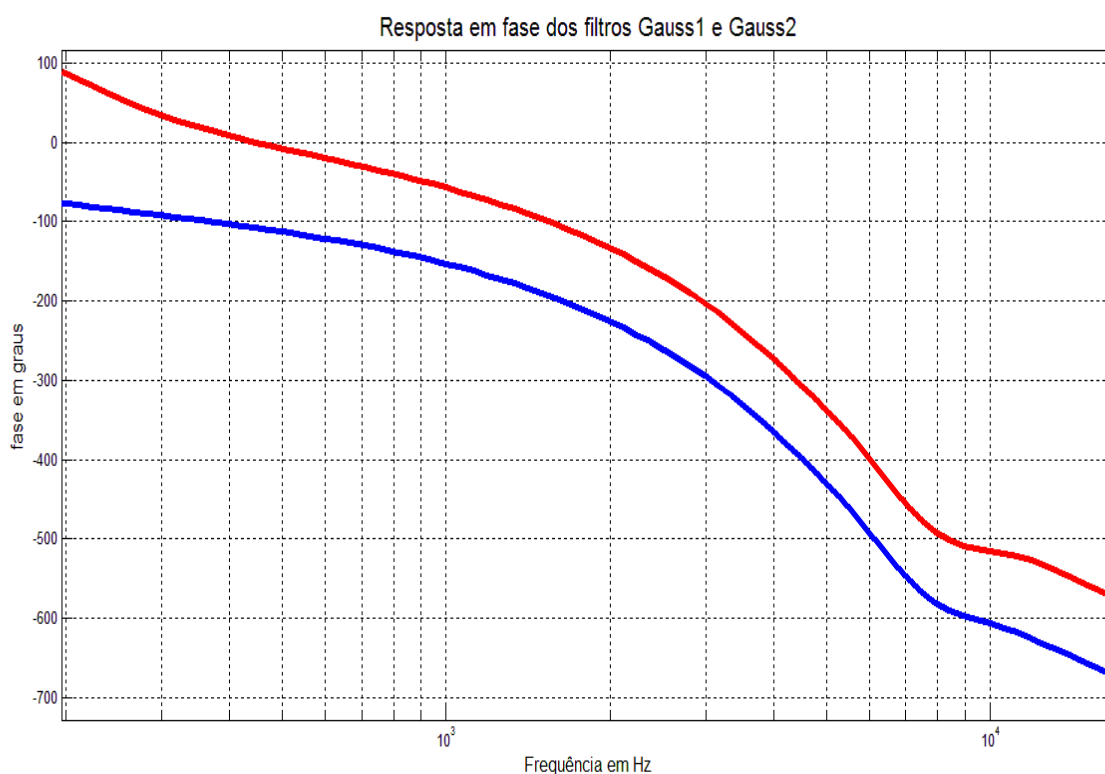


Figura 7.19 Resposta de fase em frequência dos filtros *wavelet*. Em azul vemos a resposta do filtro *gauss1* e em vermelho a resposta do filtro *gauss2*.

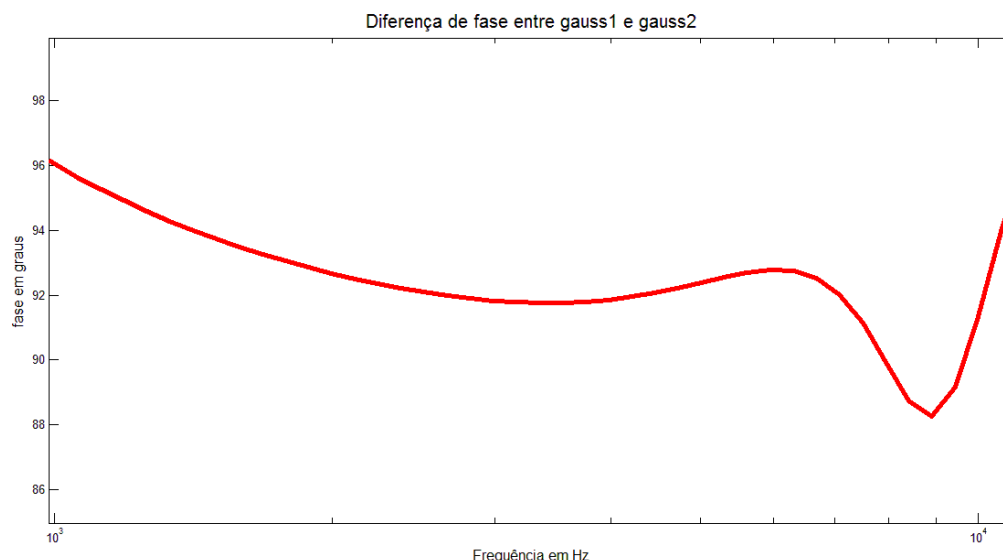


Figura 7.20 Diferença de fase entre os filtros *gauss1* e *gauss2*

Outro problema encontrado foi na diferença de fase entre as escalas das wavelets utilizadas. Conforme vemos na figura 7.20, que destaca a diferença entre as fases de duas escalas do filtro *gauss1* ao longo da frequência, percebemos que o resultado está muito longe de ser constante conforme o esperado. Diferente do modelo utilizado em algoritmo as wavelets apresentadas aqui não são simétricas com relação a um eixo de simetria, apresentando atrasos de tempo crescentes ao longo das escalas, conforme notado em (José Alberto, 2014). Uma compensação deste atraso de tempo pode ser uma solução para a minimização deste problema, porém com a variação do atraso de fase do circuito ao longo do tempo vemos que pode não ser tão simples. Vemos na figura 7.21 atrasos variantes de 60 a 200 graus dentro da banda de 1K Hz a 10K Hz.

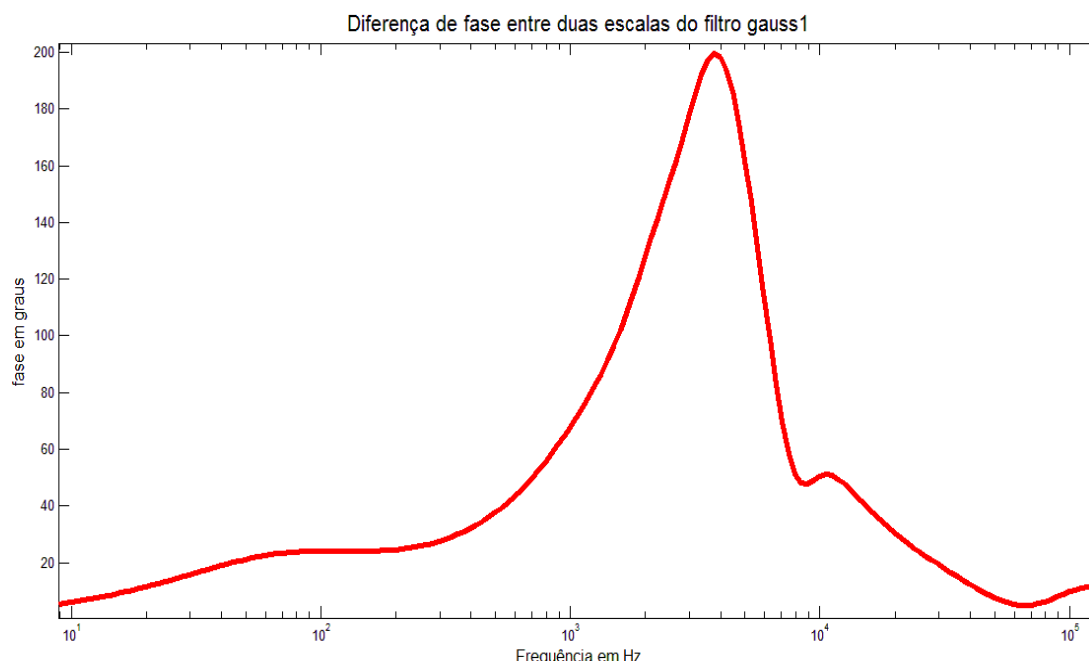


Figura 7.21 Variação da diferença de fases entre duas escalas do filtro wavelet *gauss1*.

7.5 Discussão

Por meio de um processamento de sinais antes proposto por (Mallat, 1999), conseguimos extrair características que definem o comportamento de funções em intervalos específicos, o expoente de *Lipschitz*. O uso de tal expoente nos permitiu desenvolver uma técnica nova de compressão de sinais que permite sua utilização para auxílio na conversão A/D.

Os resultados apresentados pelo conversor a níveis algorítmicos mostraram um desempenho comparável a modelos ideais conversores de alta resolução e taxa de amostragem muito elevada. Em especial a figura de mérito que relata o erro médio quadrático, em porcentagem, nos mostrou um desempenho superior a conversores de precisão média trabalhando a níveis de amostragem que podem ser impraticáveis dependendo do sinal de entrada.

Importante lembrar que os conversores de alto desempenho que foram utilizados para a comparação com o método aqui proposto podem chegar a dissipar potência de

forma impraticável para diversas aplicações, como aparelhos médicos, tais como aparelhos auditivos e marcapassos. A topologia aqui proposta propõe que a amostragem seja feita apenas em pontos críticos do sinal, o que nos leva na maior parte dos casos a frequências próximas ou pouco superiores à de *nyquist*, porém apenas pontos importantes do sinal são convertidos. Se a energia para a conversão de um sinal for proporcional a taxa de amostragem, podemos afirmar com segurança que o método proposto pode apresentar consumos muitas vezes inferior ao dos conversores com os quais foi comparado.

Apesar dos resultados satisfatórios a nível computacional, o sistema a nível de circuito precisa ser melhor trabalhado para funcionar de forma adequada. Os problemas de fase apresentados até o momento tornam o circuito impraticável se não forem bem trabalhados. Em trabalhos anteriores, foi proposta a utilização de um filtro gauss0 para a solução do problema de defasagem, uma vez que dentro da banda o mesmo deve estar constantemente a 90 graus defasado do filtro gauss1 e 180 graus do filtro gauss2 dando o comportamento desejado para a amostragem utilizada. Tal solução, porém, pode não ser boa o suficiente para o cálculo do expoente de lipschitz uma vez que o mesmo é calculado em cima das escalas da mesma wavelet e necessita que ambas estejam em fase.

8 Conclusão

O trabalho aqui exposto teve como principal objetivo encontrar métodos que permitam a implementação de conversores A/D de resolução média/elevada e baixo consumo. Características que muitas vezes se mostram em trade-off, uma vez que conversores de resolução elevada trabalham em níveis de amostragem elevados e como consequência elevam o consumo de potência consideravelmente.

No modelo proposto, para chegar ao resultado desejado, utilizou-se bancos de filtros *Wavelet* que nos permitiram amostrar pontos importantes para a recuperação de um sinal, tais como seus pontos de máximo e mínimo e singularidades. Essa recuperação se mostra possível graças à propriedade de *vanishing moments* da transformada. Além da amostragem de pontos importantes, essa mesma propriedade permitiu obter o comportamento do sinal na vizinhança do ponto por meio do expoente de *Lipschitz*.

O modelo proposto no trabalho mostrou que é possível atingir essas duas especificações, elevada resolução a uma baixa taxa de amostragem (implica consumo reduzido), graças a característica do expoente de *Lipschitz* que nos permite aproximar o comportamento de funções em uma vizinhança.

A nível de circuito, mostramos a possibilidade de se implementar filtros wavelet analógicos e que há de fato a possibilidade de se utilizar desses para realizar tanto a amostragem quanto a extração dos coeficientes de *lipschitz*. Todavia, apresentamos algumas das problemáticas que apareceram ao se tentar obter os dados necessários para realizar a conversão, sendo a maior parte dos mesmos relacionados à atrasos de fase. A nível de algoritmo podemos notar a capacidade do mesmo de recuperar sinais reais com fidelidade, notando a compressão que o mesmo fornece diminuindo a quantidade de informação necessária para que o mesmo seja reconstruído.

Para trabalhos futuros, temos a ideia de melhorias do algoritmo de reconstrução para se conseguir melhorar figuras de mérito como a relação sinal ruído e a distorção harmônica que, notando os gráficos do espectro de frequências, se mostrou superior. Vemos grande potencial deste algoritmo para processos de compressão e por esse motivo consideramos importante essa melhoria. Utilizar diferentes filtros para a

equalização de fase entre os filtros wavelet, e o então desenvolvimento de circuitos para o cálculo do coeficiente de *lipschitz* para então avaliarmos o desempenho do conversor desenvolvido com outros métodos utilizados são outras sugestões para trabalhos futuros.

ReferênciasBibliográficas

Allen, P.E. and Holberg,D.R., 2002, “CMOS Analog Circuit Design”, 2nd Ed., Ed. Oxford University Press, New York, USA, 784p

Mallat, S.,1999,”A *Wavelet* Tour of Signal Processing” 2nd Ed, Ed Academic Press

Anton H. and Horres, C., 2001, “Álgebra Linear com aplicações”, 8^a Ed, Ed. Bookman, Porto Alegre, Brasil

Haddad, S. A. P., Serdijn, W. A., 2009, “Ultra Low-power Biomedical Signal Processing: Na Analog *Wavelet* Filter Approach for Pacemarkers” Ed. Springer. 215p

Van de Plassche, R. J., 2003, “CMOS Integrated Analog-To-Digital And Digital-To-Analog Converters”, 2nd Ed., Ed. Kluwer Academic Publishers, 588p

Pelgrom, M. J. M., 2010, “Analog-To-Digital Conversion”, 2nd Ed., Ed. Springer, 584p

Mertins, A., 1999, “Signal Analysis, *Wavelets*, Filter banks, Time Frequency Transforms and Applications”, 1sr Ed., Ed. John Wiley & Sons, 317p

Gellen, G., 2001, “A 6b 1.1Gsample/s CMOS A/D converter”, IEEE International Solid-State Circuits conference, Digest of Technical Papers, vol XLIV, pp. 128-129.

Roberts, G. W., Leung, V. W., 2002, “Design and Analysis of integrator based log-domain filter circuits”, 1st Ed., Ed. Kluwer Academic Publishers

Adams, R. W., 1979, “Filtering in the log-domain”, 63rd AES conference.

Chasta, N. K., “High speed, Low power current comparators with hysteresis”, International Journal of VLSI & Communication Systems, Vol 3, No 1, 85p

Andrade, J. A. A., “Projeto de um conversor A/D, de baixo consumo para aplicações de alta resolução”, 2015.

<http://eleceng.dit.ie/dorran/matlab/ecg.txt>, Dados para teste de sinal de ECG.